

# Structured Sparse Method for Hyperspectral Unmixing

Feiyun Zhu, Ying Wang, Shiming Xiang, Bin Fan, and Chunhong Pan

## Abstract

Hyperspectral Unmixing (HU) has received increasing attention in the past decades due to its ability of unveiling information latent in hyperspectral data. Unfortunately, most existing methods fail to take advantage of the spatial information in data. To overcome this limitation, we propose a Structured Sparse regularized Nonnegative Matrix Factorization (SS-NMF) method from the following two aspects. First, we incorporate a graph Laplacian to encode the manifold structures embedded in the hyperspectral data space. In this way, the highly similar neighboring pixels can be grouped together. Second, the lasso penalty is employed in SS-NMF for the fact that pixels in the same manifold structure are sparsely mixed by a common set of relevant bases. These two factors act as a new structured sparse constraint. With this constraint, our method can learn a compact space, where highly similar pixels are grouped to share correlated sparse representations. Experiments on real hyperspectral data sets with different noise levels demonstrate that our method outperforms the state-of-the-art methods significantly.

## Index Terms

Hyperspectral Unmixing (HU), Hyperspectral Image Analysis, Structured Sparse NMF (SS-NMF), Mixed Pixel, Nonnegative Matrix Factorization (NMF).

## I. INTRODUCTION

Hyperspectral imaging has been widely used in many fields since it provides the ability to record the same scene with hundreds of contiguous and narrow spectral bands [1], [2]. Due to the low spatial resolution of the sensor, the spectra of spatial neighboring substances inevitably merge together, leading to mixed pixels in hyperspectral data. This makes Hyperspectral Unmixing (HU) an essential step for hyperspectral image analysis. Generally, the task of HU is to decompose each pixel spectrum into a set of constituent spectra (called *endmembers*, such as tree, water, soil, etc.), and their corresponding percentages (called *abundances*) [3], [4]. In this paper, we focus on the unsupervised HU problem, for which both *endmembers* and *abundances* are unknown, making HU a very challenging problem.

In general, the linear HU methods can be roughly classified into two categories: geometrical methods [5], [6], [7] and statistical ones [8], [9], [10], [11]. Geometrical methods assume that hyperspectral pixels are located within a simplex, whose vertices correspond to the *endmembers*. N-FINDR [12] and Vertex Component Analysis (VCA) [5] are two typical geometrical methods. The former treats the vertices of a simplex with the maximum volume as *endmembers*, while the latter searches the *endmembers* through projection. The above two methods are both simple and fast. Unfortunately, they require the existence of pure pixels for each *endmember*, which is usually unavailable in practice. Moreover, geometrical methods fail to consider local structures latent in hyperspectral data, leading to an inaccurate estimation of *endmembers* and *abundances*.

Feiyun Zhu, Ying Wang, Shiming Xiang, Bin Fan and Chunhong Pan are with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences (e-mail: {fyzhu, ywang, smxiang, bfan and chpan}@nlpr.ia.ac.cn).

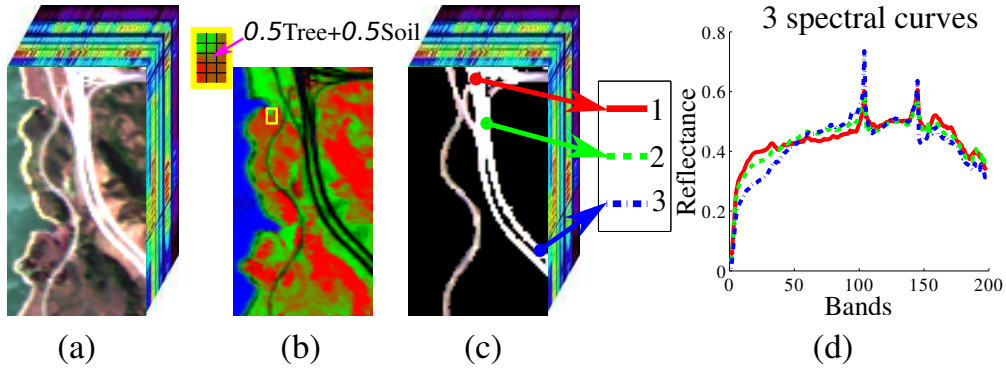


Figure 1. Illustration of two priors. (a) Hyperspectral image. (b) *Abundances* of 4 targets, which are presented by proportions of 4 colors: Tree (Red), Water (Blue), Soil (Green) and Road (Black). For example, there are 15 mixed pixels in the yellow box patch; the color proportion of the pixel pointed by arrow is  $0.5\text{Red (Tree)}+0.5\text{Green (Soil)}$ . (c) Road structure latent in data. (d) Spectral curves of pixels-1, 2, 3 are similar in the road structure. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Accordingly, a number of statistical methods have been proposed. Nonnegative Matrix Factorization (NMF) [13] and its extensions are the typical statistical methods. As an unsupervised method, NMF tries to find two nonnegative matrices to approximate the original matrix with their product. The nonnegative constraint on both factor matrices is valuable for two reasons. First, the *endmembers* and their corresponding *abundances* are both nonnegative, which makes NMF physically suited to the HU problem. Second, the nonnegative constraint allows additive, not subtractive, combinations, leading to a parts-based representation [13]. The parts-based property makes representation results more intuitive and interpretable, since psychological and physiological evidences have shown human brain works in a parts-based way [14], [15].

Although NMF has enjoyed a great success in many applications, such as face analysis [16], [17] and documents clustering [18], [19], there are three weaknesses. First, the solution space of NMF is large, which is caused by the non-convex objective function. Many extensions have been proposed to reduce the solution space by adding various constraints to the NMF framework, such as non-smooth NMF [20], MVC-NMF [21], MDMD-NMF [22], GL-NMF [23] and so on. Second, the parts-based property of NMF is usually not strong enough for the HU problem, resulting in a less expressive estimation of *endmembers*. A possible solution is to employ a sparse constraint to the NMF framework [24], [25], with a regularization parameter to control the weight of parts-based property. Third, NMF method does not consider similarities between neighboring pixels, leading to structure information unused.

In this paper we propose an effective method, named Structured Sparse NMF (SS-NMF), to overcome the above three limitations of NMF. The motivation is based on two priors. First, due to the smoothness of local patches in images, there are many regions where the spectral curves of all pixels should be similar with each other. Fig. 1c illustrates an example, where the pixels in the road region have similar visual appearance (see Fig. 1d). These regions are the structures that should be preserved while unmixing. Second, most pixels in hyperspectral data are mixed by only a few *endmembers* [25], [26], [24]. In summary, the pixels in the same structure are sparsely mixed by a common set of relevant *endmembers*. This can be obviously observed from Fig. 1b, where the proportions of four colors represent the *abundances* of four *endmembers*: Red (Tree), Blue (Water), Green (Soil) and Black (Road). As can be seen, the colors in Fig. 1b are regional smooth and sparsely mixed by Red, Blue, Green and Black ink.

Based on the above analysis, we introduce the Structured Sparse constraint (SS-constraint in short) from the following two aspects. First, we model the latent structures by encoding a weighted graph on the hyperspectral

pixels. In this way, the highly related neighboring pixels are grouped on the graph. Accordingly, a graph constraint is employed in SS-NMF to transfer these structures into the unmixing results. Second, we use the lasso penalty to sparsely encode the *abundances* for each pixel. With the SS-constraint, we can learn an *abundance* space, in which *abundances* are regional smooth and sparse, as shown in Fig. 1b. Experiments on real hyperspectral data with different noise levels illustrate that SS-NMF outperforms the state-of-the-art methods.

Below are some remarks about our method:

- We propose a meaningful SS-constraint and apply it in the NMF framework. The basic idea is to encourage highly similar pixels on the graph to share correlated sparse *abundances*, which is similar with [27]. With the help of SS-constraint, SS-NMF overcomes the three limitations of NMF for the HU problem.
- SS-NMF is an effective method for HU problems. We compare our method with several state-of-the-art methods on two hyperspectral data. Both quantitative and qualitative performances show that our method outperforms the state-of-the-art methods significantly.

The remainder of this paper is organized as follows. In Section 2, we briefly review Linear Mixture Model (LMM) and Nonnegative Matrix Factorization (NMF) method. Section 3 presents the formulation, the optimization algorithm, the convergence proof and the analyzes of computation complexity for SS-NMF method. Extensive results as well as detailed comparisons and analyses are presented in Section 4, followed by conclusions in Section 5.

## II. PRELIMINARIES

In this section, we introduce the Linear Mixture Model (LMM) and the NMF method. The former is the fundamental model for the methods mentioned in the following sections, while the latter is the basic method we build on.

### A. Linear Mixture Model (LMM)

LMM is the most popular model for hyperspectral image analysis [28], [3]. Suppose we are given a hyperspectral image with  $L$  bands and  $N$  pixels  $\{\mathbf{y}_n\}_{n=1}^N \in \mathbb{R}_+^L$ . LMM assumes that each pixel  $\mathbf{y}$  can be approximated by a nonnegative linear combination of  $K$  *endmembers* (bases) as follows

$$\mathbf{y} = \mathbf{M}\mathbf{a} + \mathbf{e}, \quad (1)$$

where  $\mathbf{M} \equiv [\mathbf{m}_1, \dots, \mathbf{m}_K] \in \mathbb{R}_+^{L \times K}$  is the *endmember* matrix with  $K$  *endmembers*,  $\mathbf{a} \in \mathbb{R}_+^K$  is the *abundance* vector associated with pixel  $\mathbf{y}$ ,  $\mathbf{e} \in \mathbb{R}^L$  is a residual term. For all pixels together, we have the matrix form of LMM as

$$\mathbf{Y} = \mathbf{M}\mathbf{A} + \mathbf{E}, \quad (2)$$

where  $\mathbf{Y} \equiv [\mathbf{y}_1, \dots, \mathbf{y}_N] \in \mathbb{R}_+^{L \times N}$  is a hyperspectral image with column vectors corresponding to pixels,  $\mathbf{A} \equiv [\mathbf{a}_1, \dots, \mathbf{a}_N] \in \mathbb{R}_+^{K \times N}$  is the *abundance* matrix, whose column vector  $\mathbf{a}_n \equiv [A_{1n}, \dots, A_{Kn}]^T \in \mathbb{R}_+^K$  contains fractional *abundances* of  $K$  *endmembers* for pixel  $\mathbf{y}_n$ .  $\mathbf{z}_k \equiv [A_{k1}, \dots, A_{kN}] \in \mathbb{R}_+^{1 \times N}$  is the  $k$ -th row vector in  $\mathbf{A}$  that contains fractional *abundances* of the  $k$ -th *endmember* (i.e.  $\mathbf{m}_k$ ) associated with all pixels. Here  $\mathbf{z}_k$  is called the  $k$ -th *abundance* map.  $\mathbf{E} \in \mathbb{R}^{L \times N}$  is a residual term.

For HU problems, we have  $K \ll \min(L, N)$ , which intimates very few *endmembers* are latent in the hyperspectral data space. In addition, most pixels are mixed by parts of the  $K$  *endmembers*. Therefore, good approximations can be achieved if the following two conditions are satisfied: (1) they can unveil intrinsic structures in hyperspectral data [13], whose rank is  $K$ ; (2) the learned *endmembers* are expressive [16].

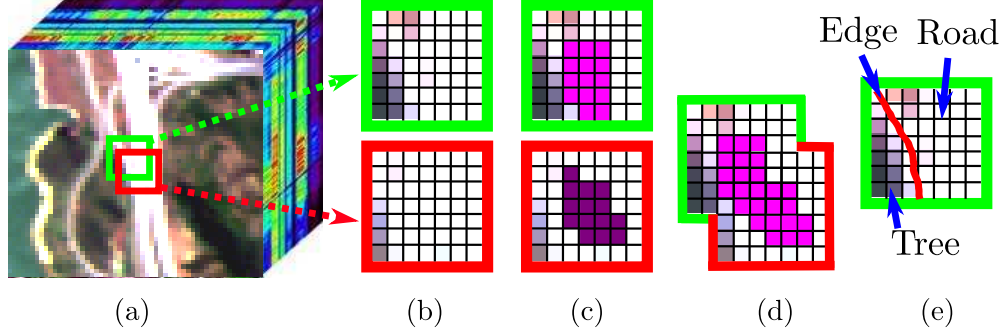


Figure 2. Highly similar neighboring pixels could be grouped by local structures. (a) A hyperspectral image. (b) Two local windows in (a). (c) The highly similar neighboring pixels are grouped together. (d) When the local structures are combined, the highly similar pixels in the two windows are grouped together. (e) Window covers an edge, where ‘Tree’ and ‘Road’ are on the two sides.

### B. Nonnegative Matrix Factorization

NMF is a popular method that builds in terms of LMM. The aim is to approximately decompose a large nonnegative matrix  $\mathbf{Y}$  into two nonnegative matrices  $\mathbf{M}$  and  $\mathbf{A}$  by minimizing the following objective function [13],

$$\min_{\mathbf{M}, \mathbf{A}} \frac{1}{2} \|\mathbf{Y} - \mathbf{MA}\|_F^2 \quad \text{s.t.} \quad \mathbf{M} \geq 0, \mathbf{A} \geq 0, \quad (3)$$

where  $\|\cdot\|_F$  is the *Frobenius* norm, all elements in  $\mathbf{M} \in \mathbb{R}_+^{L \times K}$  and  $\mathbf{A} \in \mathbb{R}_+^{K \times N}$  are unknown and nonnegative. Although (3) is convex with respect to  $\mathbf{M}$  and  $\mathbf{A}$  respectively, it is non-convex for both variables together [29]. Therefore it is unrealistic to find global minima. Alternately, [29] have proposed the multiplicative update rules as follows

$$M_{lk} \leftarrow M_{lk} \frac{(\mathbf{YA}^T)_{lk}}{(\mathbf{MAA}^T)_{lk}}, \quad A_{kn} \leftarrow A_{kn} \frac{(\mathbf{M}^T \mathbf{Y})_{kn}}{(\mathbf{M}^T \mathbf{MA})_{kn}}, \quad (4)$$

which have been proved to be non-increasing. There are other methods to solve the problem (3), such as the active-set [30] method, the alternation nonnegative least squares [31], the projected gradient descent [32] and so on.

Compared with the Vector Quantization (VQ) [33] method and the Principal Component Analysis (PCA) [34] method, the nonnegative constraint on both factor matrices enables a parts-based representation, which makes NMF superior to PCA and VQ in facial image analyses and document representations [13].

## III. STRUCTURED SPARSE NMF (SS-NMF)

As mentioned before, NMF applies the nonnegative constraint on the factor matrices, which leads to a parts-based representation of the original data. However, it fails to take consideration of the sparse prior on the *abundances* and the manifold structures hidden in hyperspectral data, which are essential for the HU problem. In this section, we introduce the SS-NMF method to overcome these limitations by adopting a structured sparse constraint.

### A. Formulation for SS-NMF

In the framework of SS-NMF, the fundamental problem is how to construct the structures. The manifold learning theory [35] and spectral graph theory [36], [37] provide us with a good idea that local structures can be learned by encoding similarities between pixels on the  $p$  nearest neighbor graph. Combining the local overlapping structures, we can get the graph structure that contains full similarity information between pixel pairs.

Typically, there are two ways to define the  $p$  nearest neighbors: considering the feature distances only and the spatial distances only. The former is well suited for less geometrically intuitive databases that treat each image as a node, such as the Yale database<sup>1</sup> and the ORL database<sup>2</sup> [35]; While the latter is adapted to geometrically motivated databases that treat each pixel, which are on the grids of a 2-D plane, as nodes, such as the hyperspectral images. Due to the local smoothness prior of the hyperspectral images, the spatially neighboring pixels are very likely to enjoy similar features (i.e. spectral signatures). However, this prior does not hold for image edges, where pixels on the two sides are dissimilar. This motivates the weighting method for the proposed SS-NMF, which would be thoroughly introduced in the following paragraph.

The proposed weighting method considers the spatial distance and the feature distance simultaneously when constructing the local structures. Suppose we are given a hyperspectral image  $\mathbf{Y} \in \mathbb{R}_+^{L \times N}$  with  $N$  pixels. We view each pixel as one node and construct a weighted graph on the  $N$  nodes. The weight between nodes  $\mathbf{y}_i$  and  $\mathbf{y}_j$  is obtained by the Spectral Angle Distance (SAD) metric [38], [5] defined as follows,

$$W_{ij} = \begin{cases} \text{SAD}(\mathbf{y}_i, \mathbf{y}_j), & \mathbf{y}_j \in \mathcal{N}(\mathbf{y}_i) \text{ or } \mathbf{y}_i \in \mathcal{N}(\mathbf{y}_j) \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

where

$$\text{SAD}(\mathbf{y}_i, \mathbf{y}_j) = \arccos \left( \frac{\mathbf{y}_i^T \mathbf{y}_j}{\|\mathbf{y}_i\| \cdot \|\mathbf{y}_j\|} \right). \quad (6)$$

In (5), the neighbors of  $\mathbf{y}_i$  are collected in  $\mathcal{N}(\mathbf{y}_i)$ , and there are two conditions for pixel  $\mathbf{y}_j \in \mathcal{N}(\mathbf{y}_i)$ : (1) nearest spatial distance, i.e.,  $\mathbf{y}_j$  is in the  $m \times m$  ( $m = 7$  in this paper) local window of  $\mathbf{y}_i$  (see Fig. 2b); (2) nearest feature distance, i.e., calculating the SAD similarities between  $\mathbf{y}_i$  and its neighboring pixels in the  $m \times m$  local window,  $\text{SAD}(\mathbf{y}_i, \mathbf{y}_j)$  is among the top 30% (empirical threshold), as shown in Fig. 2c. As a result, we can partly avoid spreading the graph across dissimilar pixels. This is essential when the local window covers the edges, where pixels on the two sides belonging to different targets. Fig. 2e illustrates an example, where ‘Road’ is on one side of the edge, while ‘Tree’ is on the other side. The SAD metric, defined in Eq. (6), is a suitable metric, since it is an essential estimation metric for *endmembers* [25], [38], [5], [26].

Through the above process, we can build the local weighted graph, or structure, with node  $\mathbf{y}_i$  at the center. In this way, the highly similar neighboring pixels are grouped together (see Figs. 2c and 2d). Repeating the above procedures for each pixel, we can construct the weight matrix  $\mathbf{W}$  that contains the manifold structures latent in hyperspectral data. Our goal is to transfer these structures to the learned *abundance* space. This could be done by first measuring the distance between *abundance* vectors  $\mathbf{a}_i$  and  $\mathbf{a}_j$

$$d(\mathbf{a}_i, \mathbf{a}_j) = \|\mathbf{a}_i - \mathbf{a}_j\|_2^2, \quad (7)$$

where  $\|\cdot\|_2$  denotes the  $\ell_2$ -norm of a vector, measuring the Euclidean distance. One intuitive way is to make  $d(\mathbf{a}_i, \mathbf{a}_j)$  small, if  $\mathbf{y}_i$  and  $\mathbf{y}_j$  are similar [35]. This can be achieved by

$$\mathcal{R} = \sum_{i=1}^N \sum_{j=1}^N d(\mathbf{a}_i, \mathbf{a}_j) W_{ij} = \text{Tr}(\mathbf{A} \mathbf{D} \mathbf{A}^T) - \text{Tr}(\mathbf{A} \mathbf{W} \mathbf{A}^T) = \text{Tr}(\mathbf{A} \mathbf{L} \mathbf{A}^T), \quad (8)$$

where  $\text{Tr}(\cdot)$  represents the trace of a matrix and  $\mathbf{D}$  is a diagonal matrix whose elements are column (or row, as  $\mathbf{W}$

<sup>1</sup><http://cvc.yale.edu/projects/yalefaces/yalefaces.html>

<sup>2</sup><http://www.uk.research.att.com/facedatabase.html>

is symmetric) sums of  $\mathbf{W}$ , i.e.,  $D_{ii} = \sum_{j=1}^N W_{ji}$ .  $\mathbf{L} = \mathbf{D} - \mathbf{W}$  is called graph Laplacian [36], [39].

By minimizing Eq. (8), the manifold structures in the original data space are transferred to the learned *abundance* space. However, Eq. (8) has not considered the fact that most manifold structures in hyperspectral data are dominated by specific sets of only a few *endmembers*. Here a lasso penalty [40], [41] on the *abundance* matrix is adopted, resulting in a structured sparse constraint as

$$\mathcal{R} = \frac{\lambda}{2} \sum_{i=1}^N \sum_{j=1}^N \mathbf{W}_{ij} \|\mathbf{a}_i - \mathbf{a}_j\|_2^2 + \alpha \sum_{n=1}^N \|\mathbf{a}_n\|_1 \quad (9)$$

$$= \frac{\lambda}{2} \text{Tr}(\mathbf{A} \mathbf{L} \mathbf{A}^T) + \alpha \|\mathbf{A}\|_1, \quad (10)$$

where  $\|\mathbf{A}\|_1 = \sum_{n=1}^N \sum_{k=1}^K |A_{kn}|$  is used to achieve an expressive set of *endmembers* [16]. Note that the structure regularization defined in Eq. (8) provides the lasso penalty with a structure constraint, under which the sparse *abundance* vectors related to the same structure tend to be similar. For example,  $\mathbf{y}_i$  and  $\mathbf{y}_j$  are in the same local structure (i.e.,  $W_{ij}$  is big), then their corresponding sparse *abundances*  $\mathbf{a}_i$  and  $\mathbf{a}_j$  tend to be similar. In addition,  $W_{ij}$  controls how strongly  $\mathbf{a}_i$  and  $\mathbf{a}_j$  share similar *abundances*. This renders the structured sparse constraint defined in Eq. (10).

Given a hyperspectral data  $\mathbf{Y} \in \mathbb{R}_+^{L \times N}$ , similar to the NMF method, the objective function of SS-NMF is defined as follows

$$\mathcal{O} = \frac{1}{2} \|\mathbf{Y} - \mathbf{M} \mathbf{A}\|_F^2 + \frac{\lambda}{2} \text{Tr}(\mathbf{A} \mathbf{L} \mathbf{A}^T) + \alpha \|\mathbf{A}\|_1, \quad (11)$$

where  $\lambda \geq 0, \alpha \geq 0$  are regularization parameters that control the complexity of the model, all elements in  $\mathbf{M}$ ,  $\mathbf{A}$  are nonnegative and unknown. We will present the optimization method to solve the problem defined in Eq. (11) in the following subsection.

### B. Optimization for SS-NMF

Similar to the NMF problem, the objective function (11) is non-convex for  $\mathbf{M}$  and  $\mathbf{A}$  together. An iterative algorithm, which could reach local minima, is introduced in this subsection. When considered the nonnegative constraint on  $\mathbf{M}$  and  $\mathbf{A}$ , the objective function (11) could be reformulated as

$$\min_{\mathbf{M}, \mathbf{A}} \frac{1}{2} \|\mathbf{Y} - \mathbf{M} \mathbf{A}\|_F^2 + \frac{\lambda}{2} \text{Tr}(\mathbf{A} \mathbf{L} \mathbf{A}^T) + \alpha \sum_{k=1}^K \sum_{n=1}^N A_{kn}, \quad \text{s.t. } \mathbf{M} \geq 0, \mathbf{A} \geq 0. \quad (12)$$

Let  $\theta_{lk}, \phi_{kn}$  be the Laplacian multipliers for constraint  $M_{lk} \geq 0$  and  $A_{kn} \geq 0$  respectively, and  $\Theta = [\theta_{lk}] \in \mathbb{R}_+^{L \times K}$ ,  $\Phi = [\phi_{kn}] \in \mathbb{R}_+^{K \times N}$ . The Lagrange  $\mathcal{L}$  is given by

$$\mathcal{L} = \frac{1}{2} \|\mathbf{Y} - \mathbf{M} \mathbf{A}\|_F^2 + \alpha \sum_{k=1}^K \sum_{n=1}^N A_{kn} + \frac{\lambda}{2} \text{Tr}(\mathbf{A} \mathbf{L} \mathbf{A}^T) + \text{Tr}(\Theta \mathbf{M}^T) + \text{Tr}(\Phi \mathbf{A}^T). \quad (13)$$

We can further obtain the partial derivative of  $\mathcal{L}$  with respect to  $\mathbf{M}$  and  $\mathbf{A}$  as

$$\frac{\partial \mathcal{L}}{\partial \mathbf{M}} = \mathbf{M} \mathbf{A} \mathbf{A}^T - \mathbf{Y} \mathbf{A}^T + \Theta \quad (14)$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{A}} = \mathbf{M}^T \mathbf{M} \mathbf{A} - \mathbf{M}^T \mathbf{Y} + \lambda \mathbf{A} \mathbf{L} + \alpha + \Phi. \quad (15)$$



Based on the Karush-Kuhn-Tucker conditions  $\theta_{lk}M_{lk} = 0$  and  $\phi_{kn}A_{kn} = 0$ , we could obtain the following equations by letting the above partial derivatives equal to zero and multiplying both sides with  $M_{lk}$  and  $A_{kn}$  for Eqs. (14) and (15) respectively,

$$(\mathbf{M}\mathbf{A}\mathbf{A}^T)_{lk}M_{lk} - (\mathbf{Y}\mathbf{A}^T)_{lk}M_{lk} = 0 \quad (16)$$

$$(\mathbf{M}^T\mathbf{M}\mathbf{A})_{kn}A_{kn} - (\mathbf{M}^T\mathbf{Y})_{kn}A_{kn} + \lambda(\mathbf{A}\mathbf{L})_{kn}A_{kn} + \alpha A_{kn} = 0. \quad (17)$$

When the equation  $\mathbf{L} = \mathbf{D} - \mathbf{W}$  mentioned before is considered, Eq. (17) could be rewritten as

$$[(\mathbf{M}^T\mathbf{M}\mathbf{A})_{kn} + \lambda(\mathbf{L}\mathbf{D})_{kn} + \alpha]A_{kn} = [(\mathbf{M}^T\mathbf{Y})_{kn} + \lambda(\mathbf{L}\mathbf{W})_{kn}]A_{kn}. \quad (18)$$

Solving Eqs. (16) and (18), we have the updating rules as

$$M_{lk} \leftarrow M_{lk} \frac{(\mathbf{Y}\mathbf{A}^T)_{lk}}{(\mathbf{M}\mathbf{A}\mathbf{A}^T)_{lk}} \quad (19)$$

$$A_{kn} \leftarrow A_{kn} \frac{(\mathbf{M}^T\mathbf{Y} + \lambda\mathbf{A}\mathbf{W})_{kn}}{(\mathbf{M}^T\mathbf{M}\mathbf{A} + \lambda\mathbf{A}\mathbf{D} + \alpha)_{kn}}. \quad (20)$$

Therefore, Eq. (12) could be solved by alternately updating  $\mathbf{M}$  and  $\mathbf{A}$  according to Eqs. (19) and (20) respectively. Our algorithm of SS-NMF is summarized in Algorithm 1.

It is worthwhile to point out that if  $\mathbf{M}$  and  $\mathbf{A}$  form the solution for NMF algorithm, then  $\mathbf{D}\mathbf{U}$  and  $\mathbf{U}^{-1}\mathbf{A}$  are the solution for any diagonal matrix  $\mathbf{U}$  with positive diagonal elements. To get rid of this uncertainty, a simple and widely used method is to scale each column of  $\mathbf{M}$  or each row of  $\mathbf{A}$  to be unit  $\ell_2$ -norm [18] (or  $\ell_1$ -norm). This can be achieved by,

$$M_{lk} \leftarrow \frac{M_{lk}}{\sqrt{\sum_{l=1}^L M_{lk}^2}}, \quad A_{kn} \leftarrow A_{kn} \sqrt{\sum_l M_{lk}^2}. \quad (21)$$

The proposed SS-NMF also employs this strategy to eliminate the uncertainty, which is essential in the sense of computer realization.

---

#### Algorithm 1 for SS-NMF

---

**Input:** Hyperspectral data  $\mathbf{Y}$ , the number of *endmembers*  $K$ , parameters  $\lambda$ ,  $\alpha$ , and threshold  $\tau$ .

**Output:** *Endmember* matrix  $\mathbf{M} \in \mathbb{R}_+^{L \times K}$  and *abundance* matrix  $\mathbf{A} \in \mathbb{R}_+^{K \times N}$

---

- 1: **Initialize** the *endmember* matrix by randomly selecting  $K$  pixels from  $\mathbf{Y}$ . Each newly selected pixel should be dissimilar with the pixels have been selected [26]. Initialize  $\mathbf{A}$  with random positive values, and scale each column of  $\mathbf{A}$  to be unit  $\ell_1$ -norm.
  - 2: Construct the weight matrix  $\mathbf{W}$  on the hyperspectral image based on Eqs. (5), (6) and calculate the degree matrix  $\mathbf{D}$  and graph laplacian matrix  $\mathbf{L}$ .
  - 3: **while**  $\frac{\|\mathcal{O}_{old} - \mathcal{O}_{new}\|}{\mathcal{O}_{old}} \geq \tau$  ( $\mathcal{O}$  defined in Eq. (11)) **do**
  - 4:   update  $\mathbf{A}$  by Eq. (20).
  - 5:   update  $\mathbf{M}$  by Eq. (19).
  - 6:   scale  $\mathbf{M}$  and  $\mathbf{A}$  by Eq. (21) after each iteration.
  - 7: **end while**
  - 8: Output  $\mathbf{M}$  and  $\mathbf{A}$  as the final unmixing result.
-

### C. Proof of Convergence

In this subsection, we demonstrate that the optimization problem Eq. (12) is non-increasing by using the updating rules (19) and (20) during each iteration, and it finally converges to local minima after finite iterations. Since the updating rule for  $\mathbf{M}$  is the same as that of NMF and the convergence proof is provided by [29], we only focus on the proof of the updating rule (20) for  $\mathbf{A}$ . A common skill used in the Expectation Maximization (EM) algorithm [42], [43] is adopted by introducing an auxiliary function, which is exactly an upper bound function.

**Definition 1.**  $G(\mathbf{A}, \mathbf{A}')$  is an auxiliary function for  $F(\mathbf{A})$  if the following properties are satisfied:

$$G(\mathbf{A}, \mathbf{A}') \geq F(\mathbf{A}), \quad G(\mathbf{A}, \mathbf{A}) = F(\mathbf{A}).$$

By minimizing the energy of  $G(\mathbf{A}, \mathbf{A}')$  exactly given by

$$\mathbf{A}^{(t+1)} = \arg \min_{\mathbf{A}} G(\mathbf{A}, \mathbf{A}^{(t)}), \quad (22)$$

we can get a suitable solution  $\mathbf{A}^{(t+1)}$  that makes  $F(\mathbf{A})$  non-increasing

$$F(\mathbf{A}^{(t+1)}) \leq F(\mathbf{A}^{(t)}). \quad (23)$$

*Proof:* This is because of the following inequalities,

$$F(\mathbf{A}^{(t+1)}) = G(\mathbf{A}^{(t+1)}, \mathbf{A}^{(t+1)}) \leq G(\mathbf{A}^{(t+1)}, \mathbf{A}^{(t)}) \leq G(\mathbf{A}^{(t)}, \mathbf{A}^{(t)}) = F(\mathbf{A}^{(t)}). \quad (24)$$

■

Now, the objective function defined in Eq. (11) is only related to variable  $\mathbf{A}$ . It could be represented by  $F(\mathbf{A})$  as

$$F(\mathbf{A}) = \frac{1}{2} \|\mathbf{Y} - \mathbf{MA}\|_F^2 + \frac{\lambda}{2} \text{Tr}(\mathbf{ALA}^T) + \alpha \|\mathbf{A}\|_1. \quad (25)$$

Specifically, it is a quadratic function that equals to the Taylor expansion

$$F(\mathbf{A}) = F(\mathbf{A}^{(t)}) + \text{Tr}(\mathbf{C}^T \nabla F(\mathbf{A}^{(t)})) + \frac{1}{2} \text{Tr}(\mathbf{C}^T (\mathbf{M}^T \mathbf{M}) \mathbf{C}) + \frac{\lambda}{2} \text{Tr}(\mathbf{CLC}^T), \quad (26)$$

where  $\mathbf{C} = (\mathbf{A} - \mathbf{A}^{(t)}) \in \mathbb{R}^{K \times N}$ . A function constituted based on the updating rule (20) is given by

$$G(\mathbf{A}, \mathbf{A}^{(t)}) = F(\mathbf{A}^{(t)}) + \text{Tr}(\mathbf{C}^T \nabla F(\mathbf{A}^{(t)})) + \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N Q_{kn} C_{kn}^2, \quad (27)$$

where

$$Q_{kn} = \frac{(\mathbf{M}^T \mathbf{MA}^{(t)} + \lambda \mathbf{A}^{(t)} \mathbf{D} + \alpha)_{kn}}{A_{kn}^{(t)}}. \quad (28)$$

This means that we could obtain the updating rule (20) by minimizing the function  $G(\mathbf{A}, \mathbf{A}^{(t)})$  at the  $t$ -th iteration step.

**Lemma 2.** The Function  $G(\mathbf{A}, \mathbf{A}^{(t)})$  defined in (27) is an auxiliary function (i.e. upper bound function) for the function  $F(\mathbf{A})$  defined in (25).

*Proof:* When  $\mathbf{A} = \mathbf{A}^{(t)}$ , i.e.  $\mathbf{C} = \mathbf{0}$ , is satisfied, we have  $F(\mathbf{A}) = F(\mathbf{A}^{(t)}) = G(\mathbf{A}, \mathbf{A}^{(t)})$ . Specifically, when  $\mathbf{A} \neq \mathbf{A}^{(t)}$  (i.e.  $\mathbf{C} \neq \mathbf{0}$ ), we have to prove  $G(\mathbf{A}, \mathbf{A}^{(t)}) \geq F(\mathbf{A})$ . Since the constant term and linear term in Eq.(26)



are equal to their counterparts in Eq.(27), Lemma 2 could be proven by comparing the quadratic terms as

$$\sum_{k=1}^K \sum_{n=1}^N Q_{kn} C_{kn}^2 \geq \text{Tr}(\mathbf{C}^T (\mathbf{M}^T \mathbf{M}) \mathbf{C}) + \lambda \text{Tr}(\mathbf{C} \mathbf{L} \mathbf{C}^T). \quad (29)$$

Since  $\mathbf{D}$  is a diagonal matrix, Eq. (28) is simplified as

$$Q_{kn} = \frac{(\mathbf{M}^T \mathbf{M} \mathbf{A}^{(t)})_{kn}}{A_{kn}^{(t)}} + \lambda D_{nn} + \frac{\alpha}{A_{kn}^{(t)}}. \quad (30)$$

The inequality (29) could be rewritten as

$$\underbrace{\left\{ \sum_{k,n} \frac{(\mathbf{M}^T \mathbf{M} \mathbf{A}^{(t)})_{kn}}{A_{kn}^{(t)}} C_{kn}^2 - \text{Tr}(\mathbf{C}^T \mathbf{M}^T \mathbf{M} \mathbf{C}) \right\}}_{\text{first term}} + \underbrace{\left\{ \sum_{k,n} \left( \lambda D_{nn} + \frac{\alpha}{A_{kn}^{(t)}} \right) C_{kn}^2 - \lambda \text{Tr}(\mathbf{C} \mathbf{L} \mathbf{C}^T) \right\}}_{\text{second term}} \geq 0, \quad (31)$$

which could be proven by proving both terms to be greater than or equal to zero.

To begin with, the first term in Eq. (31) is compared as

$$\begin{aligned} f_1 &= \sum_{k=1}^K \sum_{n=1}^N \frac{(\mathbf{M}^T \mathbf{M} \mathbf{A}^{(t)})_{kn}}{A_{kn}^{(t)}} C_{kn}^2 - \text{Tr}(\mathbf{C}^T \mathbf{M}^T \mathbf{M} \mathbf{C}) \\ &= \sum_{k=1}^K \sum_{n=1}^N \frac{(\mathbf{M}^T \mathbf{M} \mathbf{A}^{(t)})_{kn}}{A_{kn}^{(t)}} C_{kn}^2 - \sum_{k=1}^K \sum_{l=1}^K \sum_{n=1}^N C_{kn} C_{ln} (\mathbf{M}^T \mathbf{M})_{lk} \\ &= \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N \sum_{l=1}^K (\mathbf{M}^T \mathbf{M})_{kl} \frac{\left( A_{ln}^{(t)} \right)^2 C_{kn}^2 + \left( A_{kn}^{(t)} \right)^2 C_{ln}^2 - 2 A_{ln}^{(t)} A_{kn}^{(t)} C_{kn} C_{ln}}{A_{kn}^{(t)} A_{ln}^{(t)}} \\ &= \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N \sum_{l=1}^K (\mathbf{M}^T \mathbf{M})_{kl} \frac{\left( A_{ln}^{(t)} C_{kn} - A_{kn}^{(t)} C_{ln} \right)^2}{A_{kn}^{(t)} A_{ln}^{(t)}} \geq 0 \end{aligned} \quad (32)$$

Then we compare the second term in Eq. (31)

$$f_2 = \sum_{k=1}^K \sum_{n=1}^N \left( \lambda D_{nn} + \frac{\alpha}{A_{kn}^{(t)}} \right) C_{kn}^2 - \lambda \text{Tr}(\mathbf{C} \mathbf{L} \mathbf{C}^T). \quad (33)$$

When the equation  $\mathbf{L} = \mathbf{D} - \mathbf{W}$  is considered,  $\text{Tr}(\mathbf{C} \mathbf{L} \mathbf{C}^T)$  could be rewritten as

$$\text{Tr}(\mathbf{C} \mathbf{L} \mathbf{C}^T) = \sum_{n=1}^N \sum_{k=1}^K D_{nn} C_{kn}^2 - \sum_{n=1}^N \sum_{m=1}^N \sum_{k=1}^K C_{kn} C_{km} W_{mn}. \quad (34)$$

Because of the inequality,

$$\begin{aligned} \sum_{n=1}^N \sum_{m=1}^N \sum_{k=1}^K C_{kn} C_{km} W_{mn} &\geq \frac{1}{2} \sum_{n=1}^N \sum_{m=1}^N \sum_{k=1}^K W_{mn} (-C_{kn}^2 - C_{km}^2) \\ &= - \sum_{n=1}^N \sum_{k=1}^K \left( \sum_{m=1}^N W_{mn} \right) C_{kn}^2 = - \sum_{n=1}^N \sum_{k=1}^K D_{nn} C_{kn}^2, \end{aligned}$$

Eq. (34) has the following relationship,

$$\text{Tr}(\mathbf{CLC}^T) \leq 2 \sum_{n=1}^N \sum_{k=1}^K D_{nn} C_{kn}^2. \quad (35)$$

Thus Eq. (33) could be relaxed as

$$\begin{aligned} f_2 &\geq \sum_{k=1}^K \sum_{n=1}^N \left( \lambda D_{nn} + \frac{\alpha}{A_{kn}^{(t)}} \right) C_{kn}^2 - 2\lambda \sum_{n=1}^N \sum_{k=1}^K D_{nn} C_{kn}^2 \\ &= \sum_{k=1}^K \sum_{n=1}^N \left( \frac{\alpha}{A_{kn}^{(t)}} - \lambda D_{nn} \right) C_{kn}^2. \end{aligned} \quad (36)$$

For simplicity, we ignore the impact of  $C_{kn}^2$  and obtain

$$f_2' = \sum_{k=1}^K \sum_{n=1}^N \frac{\alpha}{A_{kn}^{(t)}} - \sum_{k=1}^K \sum_{n=1}^N \lambda D_{nn} \geq \frac{\alpha}{u} - KN\lambda v. \quad (37)$$

where  $u = \min \{A_{kn}^{(t)}\}$  and  $v = \max \{D_{nn}\}$  for  $n = \{1, \dots, N\}$ ,  $k = \{1, \dots, K\}$ . Because of the sparse distribution of the column vectors in  $\mathbf{A}$  and the sparse constraint on  $\mathbf{A}$ , there are always some vector  $[A_{1n}^{(t)}, \dots, A_{Kn}^{(t)}]^T$  very sparse, which means  $u \rightarrow 0$  and  $\frac{\alpha}{u} \rightarrow \infty$ . Besides, the values of  $\alpha$  and  $KN\lambda v$  are positive and finite, we have  $(\frac{\alpha}{u} - KN\lambda v) \gg 0$ . Mostly, it is straightforward that  $f_2 \geq 0$  in (36). Thus, we prove the inequality (29) by proving  $f_1 \geq 0$  in (32) and  $f_2 \geq 0$  in (36).  $G(\mathbf{A}, \mathbf{A}^{(t)})$  is an auxiliary function (upper bound function) for  $F(\mathbf{A})$ . ■

#### D. Comparison with Gradient Descent Method

The Gradient Descent Method (GDM) [44] is a widely used optimization algorithm to find a local minimum for an objective function. In this subsection, we try to find the relationship between the updating rules defined in Eqs. (19), (20) and that of Gradient Descent Method. For the problem defined in (11), the basic updating rules for GDM are given by

$$M_{lk} \leftarrow M_{lk} + u_{lk} \frac{\partial \mathcal{O}}{\partial M_{lk}}, \quad A_{kn} \leftarrow A_{kn} + v_{kn} \frac{\partial \mathcal{O}}{\partial A_{kn}}. \quad (38)$$

The parameters  $[u_{lk}] \in \mathbb{R}^{L \times K}$ ,  $[v_{kn}] \in \mathbb{R}^{K \times N}$  are the learning rates. For our problem, there are two conditions to ensure physical meaningful local minima obtained by the updating rules in (38). First, the values of  $u_{lk}$  and  $v_{kn}$  are relatively small to get local minima. Second, the values of  $u_{lk}$  and  $v_{kn}$  could ensure the nonnegative property of  $M_{lk}$  and  $A_{kn}$  during each iteration. One kind of choices to determine the learning rates  $u_{lk}$  and  $v_{kn}$  are as follows:

- Let  $u_{lk} = -M_{lk} / (\mathbf{MAA}^T)_{lk}$ , we have

$$M_{lk} + u_{lk} \frac{\partial \mathcal{O}}{\partial M_{lk}} = M_{lk} \frac{(\mathbf{YA}^T)_{lk}}{(\mathbf{MAA}^T)_{lk}}.$$

- Let  $v_{kn} = -A_{kn} / (\mathbf{M}^T \mathbf{MA} + 2\lambda \mathbf{AD} + \alpha)$ , we have

$$A_{kn} + v_{kn} \frac{\partial \mathcal{O}}{\partial A_{kn}} = A_{kn} \frac{(\mathbf{M}^T \mathbf{Y} + 2\lambda \mathbf{AW})_{kn}}{(\mathbf{M}^T \mathbf{MA} + 2\lambda \mathbf{AD} + \alpha)_{kn}}.$$

It is straightforward that Eqs. (19) and (20) are special updating rules that could be obtained by the Gradient Descent Method.

Table I  
PARAMETERS USED IN COMPUTATIONAL COMPLEXITY ANALYSIS.

| Parameters | Description   |
|------------|---|
| $L$        | the number of bands                                     |
| $K$        | the number of <i>endmembers</i>                         |
| $N$        | the number of pixels in the hyperspectral image         |
| $p$        | the number of pixels in the local window                |
| $q$        | the percentage of nearest neighbors in the local window |
| $t$        | the number of iterations                                |

Table II  
COMPUTATIONAL OPERATION COUNTS FOR NMF AND SS-NMF DURING EACH ITERATION.

|        | addition   | multiplication   | division   | overall  |
|--------|--|--|------------|----------|
| NMF    | $2LNK + \frac{3}{2}K^2(L + N) - \frac{3}{2}K(N + L) - 2K^2$              | $2LNK + \frac{3}{2}K^2(L + N) + K(L + N)$              | $K(L + N)$ | $O(LNK)$ |
| SS-NMF | $2LNK - \frac{3}{2}K(N + L) - 2K^2 + \frac{3}{2}K^2(L + N) + K(pq + 2)N$ | $2LNK + \frac{3}{2}K^2(L + N) + K(L + N) + K(pq + 1)N$ | $K(L + N)$ | $O(LNK)$ |

### E. Computational Complexity Analysis

In this subsection, we compare the computational complexity of the proposed SS-NMF with that of the NMF algorithm. Since the SS-NMF and NMF algorithms are solved in an iteration way. We describe the complexity analysis in two steps. First, we analyze the computational complexity for each iteration; Second, the number of iteration steps are considered.

Besides the updating rules, SS-NMF needs about  $O(LNp)$  to construct the structure relationships between pixel pairs. Suppose that totally  $t$  iterations are needed to get a convergence result. The total computational cost for NMF is about

$$O(tLNK). \quad (39)$$

The total computational cost for SS-NMF is about

$$O(tLNK + LNp). \quad (40)$$

As can be seen, the computational complexity of SS-NMF is a bit more than that of NMF when both methods need  $t$  iterations. However, we will show that the iterations needed by SS-NMF are less than that of NMF in subsection IV-H. In conclusion, the total computational complexity for SS-NMF is very close to that of NMF.

Usually, the big  $O$  notation [45] is used to analyze the complexity of an algorithm. To be clear, the precise arithmetic operations as well as the complexity analysis in the big  $O$  notation for each iteration are summarized in Table II. Table I lists the parameters used in the complexity analysis. There are three kinds of arithmetic operations in each iteration: addition, multiplication and division. All of them are calculated in the float-point precision. Specifically, two tips are essential to get the results in Table II. For tip 1, the order of the matrix multiplication is important. Taking the matrix multiplication  $\mathbf{M}\mathbf{A}\mathbf{A}^T$  for example, there are two orders:  $(\mathbf{M}\mathbf{A})\mathbf{A}^T$  and  $\mathbf{M}(\mathbf{A}\mathbf{A}^T)$ . The former approximately needs  $2LNK$  AM (Addition and Multiplication) and the latter costs  $(N + L)K^2$  AM (Addition and Multiplication). Since  $K \ll \min\{N, L\}$ , we have  $2LNK \gg (N + L)K^2$ , which means that  $(\mathbf{M}\mathbf{A})\mathbf{A}^T$  is much

more complex than  $\mathbf{M}(\mathbf{A}\mathbf{A}^T)$ . For tip 2, there are symmetry matrices in the computation. If the fact that  $(\mathbf{A}\mathbf{A}^T)$  is a symmetry matrix is considered, the multiplication of  $\mathbf{M}$  with  $(\mathbf{A}\mathbf{A}^T)$  costs about  $(\frac{1}{2}L + N) K^2$  AM (Addition and Multiplication).

#### IV. EXPERIMENTS

In this section, we investigate the effectiveness of the proposed SS-NMF algorithm for the Hyperspectral Unmixing (HU) problem. Several experiments are carried out to show that our algorithm is well suited for the HU problem.

##### A. Evaluation Metrics

In order to evaluate the proposed method, we adopt two performance metrics: the Spectral Angle Distance (SAD) and the Root Mean Square Error (RMSE), which are widely used by [25], [38], [26], [5]. The SAD is used to evaluate the performance of estimated *endmembers*, which is an angle distance between an estimated *endmember* and its corresponding ground truth. It is defined as follows

$$\text{SAD}(\mathbf{m}, \hat{\mathbf{m}}) = \arccos \left( \frac{\mathbf{m}^T \hat{\mathbf{m}}}{\|\mathbf{m}\| \cdot \|\hat{\mathbf{m}}\|} \right), \quad (41)$$

where  $\mathbf{m}$  denotes the ground truth of one *endmember*,  $\hat{\mathbf{m}}$  is the corresponding estimated result. The smaller SAD corresponds to a better performance. The Root Mean Square Error (RMSE) is used to evaluate the performance of estimated *abundance* maps. It is given by

$$\text{RMSE}(\mathbf{z}, \hat{\mathbf{z}}) = \left( \frac{1}{N} \|\mathbf{z} - \hat{\mathbf{z}}\|_2^2 \right)^{\frac{1}{2}}, \quad (42)$$

where  $N$  is the number of pixels in the hyperspectral image,  $\mathbf{z}$  is the ground truth of one *abundance* map,  $\hat{\mathbf{z}}$  denotes the corresponding estimated result. The smaller RMSE corresponds to a better performance. In the following subsections, the *abundance* map (i.e.  $\mathbf{z}$ ) will be showed in two visible ways: in pseudo color and in gray scale, as shown in Figs. 12, 13, 14 and 15.

##### B. Compared Algorithms

We compare the proposed method with seven related methods on several data sets. The details of these evaluated algorithms (including our algorithm) are listed as follows

- 1) **Our algorithm:** Structured Sparse regularized Nonnegative Matrix Factorization (SS-NMF in short) is a new algorithm proposed in this paper.
- 2) Vertex Component Analysis (VCA in short) [5] is a classic geometrical method that needs the existence of pure pixels for each *endmember*. Different with the other algorithms that estimate the *endmembers* and *abundances* simultaneously, VCA can only estimate the *endmembers*. The *abundances* are estimated by solving a constrained Least Square Problem [46]. The code for this algorithm is obtained from “<http://www.lx.it.pt/biucac/code.htm>”.
- 3) Nonnegative Matrix Factorization (NMF in short) [13] is a typical statistical method. Due to the nonnegative constraint, which could be seen as a special case of sparsity constraint [25], NMF tends to get parts-based results. The code for this method is downloaded from “<http://www.ee.columbia.edu/grindlay/code.html>”.
- 4) Nonnegative sparse coding ( $\ell_1$ -NMF in short) is a popular sparse regularized NMF method proposed by [47]. The code is available on “<http://www.cs.helsinki.fi/u/phoyer/software.html>”.

- 5)  $\ell_{1/2}$  sparsity-constrained Nonnegative Matrix Factorization ( $\ell_{1/2}$ -NMF in short) is a state-of-the-art algorithm for the HU problem. It is proposed by [25]. Since the code is unavailable on the Internet, we implement it by ourself.
- 6) Graph regularized Nonnegative Matrix Factorization (G-NMF in short), proposed by [39], is a good algorithm to extract graph information latent in data and transfer it to the low dimension representation space in the matrix factorization process. The code is obtained from “<http://www.cad.zju.edu.cn/home/dengcai/Data/GNMF.html>”.
- 7) Local Neighborhood Weights regularized NMF (W-NMF in short) [48] is a graph based NMF method. The main contribution of this method is that it integrates the spectral and spatial information when constructing the weighted graph. Since the code implemented by the author has been lost, we realize it by ourself.
- 8) Endmember Dissimilarity Constrained NMF (EDC-NMF in short) [49] is different from the listed methods. It is the only method that utilizes a constraint from the endmember prior. With this constraint, it would be highly possible for EDC-NMF to find a result whose endmember spectra are smooth and different from each other. The code is implemented by ourself.

Among the above eight algorithms, VCA is a geometrical method, the other seven algorithms belong to statistical ones. There is no parameter in the NMF and VCA methods. In the next subsection, the methods to set parameters for  $\ell_1$ -NMF,  $\ell_{1/2}$ -NMF, G-NMF, W-NMF and SS-NMF algorithms are introduced.

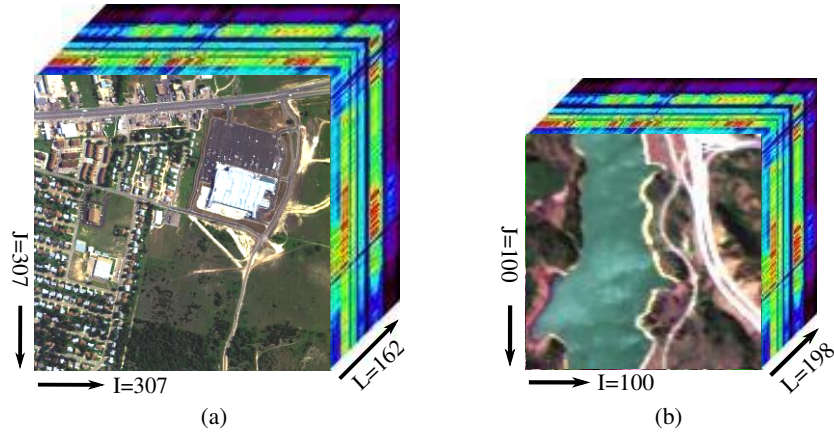


Figure 3. Real hyperspectral data sets used in the experiment: (a) Urban, with  $307 \times 307$  pixels and 162 bands; (b) Jasper Ridge, with  $100 \times 100$  pixels and 198 bands.

### C. Parameter Settings

There are two essential parameters in the proposed SS-NMF method:  $\alpha$  controls the lasso penalty and  $\lambda$  controls the structured constraint. In this subsection, we introduce the methods to set  $\alpha$  and  $\lambda$  respectively.

The value of  $\alpha$  is closely related to the sparsity of the *abundance* matrix  $\mathbf{A}$ . Unfortunately,  $\mathbf{A}$  is unknown. Here we adopt a method to estimate  $\alpha$  by calculating the sparsity value of hyperspectral data, which is based on the metric [50], [25] given by

$$\alpha_0 = \frac{1}{\sqrt{L}} \sum_{l=1}^L \frac{\sqrt{N} - \|\mathbf{x}_l\|_1 / \|\mathbf{x}_l\|_2}{\sqrt{N} - 1}, \quad (43)$$

where  $\mathbf{x}_l \in \mathbb{R}_+^N$  is the image in the  $l$ -th channel of hyperspectral data. To improve the accuracy, we set  $\alpha$  by searching the range  $[10^{-1}\alpha_0, 10\alpha_0]$  at 50 equally spaced values. The parameters in  $\ell_1$ -NMF and  $\ell_{1/2}$ -NMF are estimated in the same way.

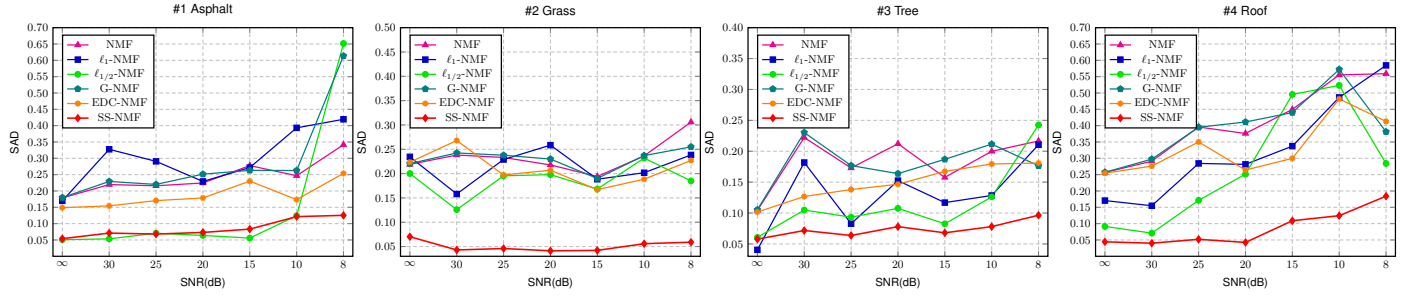


Figure 4. SADs of four *endmembers* vs. SNRs on the Urban data: (a) Asphalt, (b) Grass, (c) Tree and (d) Roof. The symbol ‘ $\infty$ ’ at the bottom-left of each subfigure indicates that no Gaussian noise is added manually.

Table III

THE AVERAGE SADs AND THEIR STANDARD DEVIATIONS VS. SNRS ON URBAN DATA. IN EACH ROW, THE VALUE MARKED IN BOLD IS THE BEST.

| SNR      | Spectral Angle Distance $\overline{\text{SAD}} (\times 10^{-2})$ |                 |                 |                   |                 |                 |                 |                                   |
|----------|--|-----------------|-----------------|-------------------|-----------------|-----------------|-----------------|-----------------------------------|
|          | VCA  | NMF             | $\ell_1$ -NMF   | $\ell_{1/2}$ -NMF | G-NMF           | W-NMF           | EDC-NMF         | SS-NMF                            |
| $\infty$ | $40.6 \pm 1.82$  | $19.0 \pm 0.10$ | $15.4 \pm 0.07$ | $10.1 \pm 0.17$   | $19.1 \pm 0.11$ | $20.7 \pm 0.14$ | $18.2 \pm 0.08$ | <b><math>5.7 \pm 0.10</math></b>  |
| 30       | $39.4 \pm 4.69$  | $24.3 \pm 0.69$ | $20.6 \pm 0.91$ | $8.9 \pm 0.26$    | $25.0 \pm 0.68$ | $28.3 \pm 0.61$ | $20.6 \pm 0.82$ | <b><math>5.6 \pm 0.01</math></b>  |
| 25       | $38.5 \pm 3.92$  | $25.4 \pm 0.08$ | $22.2 \pm 0.08$ | $13.3 \pm 0.17$   | $25.7 \pm 0.07$ | $30.7 \pm 0.25$ | $21.4 \pm 0.08$ | <b><math>5.7 \pm 0.02</math></b>  |
| 20       | $37.9 \pm 5.24$  | $25.8 \pm 0.12$ | $23.0 \pm 2.89$ | $15.5 \pm 0.23$   | $26.4 \pm 0.09$ | $39.0 \pm 0.48$ | $19.9 \pm 0.06$ | <b><math>5.9 \pm 0.01</math></b>  |
| 15       | $37.8 \pm 5.07$  | $27.0 \pm 0.07$ | $22.9 \pm 0.75$ | $20.1 \pm 0.59$   | $27.0 \pm 0.07$ | $49.3 \pm 1.35$ | $21.6 \pm 0.07$ | <b><math>7.6 \pm 0.05</math></b>  |
| 10       | $37.7 \pm 5.28$  | $31.0 \pm 0.06$ | $30.2 \pm 0.53$ | $25.1 \pm 0.87$   | $32.1 \pm 0.08$ | $44.3 \pm 0.02$ | $25.6 \pm 0.06$ | <b><math>9.5 \pm 0.06</math></b>  |
| 8        | $38.4 \pm 5.47$  | $35.6 \pm 1.77$ | $36.3 \pm 1.52$ | $34.1 \pm 0.94$   | $35.7 \pm 0.88$ | $46.2 \pm 1.21$ | $26.9 \pm 0.14$ | <b><math>11.6 \pm 0.02</math></b> |
| Avg.     | 38.6   | 26.9            | 24.4            | 18.1              | 27.3            | 36.9            | 22.0            | <b>7.4</b>                        |

One intuitive thought is that the value of  $\lambda$  is closely related to the level of similarities between pixel pairs in the neighborhood. Thus, the value of  $\lambda_0$  is estimated in two steps. First, we randomly select 100 local patches with  $5 \times 5$  pixels respectively and compute the similarities between each central pixel and its neighboring ones. Second,  $\lambda_0$  is set by averaging all these  $100 \times (5 \times 5 - 1) = 2400$  values. To improve the accuracy, we set  $\lambda$  by searching the range  $[10^{-4}\lambda_0, 10\lambda_0]$  at 50 equally spaced values. The parameter to control the strength of the graph constraint in G-NMF and W-NMF is set similarly.

#### D. Real Data Sets

We use two popular hyperspectral data sets [25], [38], [51], as shown in Fig. 3, to evaluate the proposed method. The details of these experiment data sets are introduced in this subsection.

**Urban Data**, available at <http://www.tec.army.mil/Hypercube>, is one of the most widely used hyperspectral data set for the HU research [25], [26]. It was recorded by the Hyperspectral Digital Imagery Collection Experiment (HYDICE) in October 1995, whose location is an urban area at Copperas Cove, TX, U.S. There are  $307 \times 307$  pixels in this image. Each pixel, corresponding to a  $2 \times 2 m^2$  area, is observed at 210 wavelengths ranging from 400 nm to 2500 nm, with a spectral resolution of 10 nm. After the bands 1–4, 76, 87, 101–111, 136–153 and 198–210 are removed (due to dense water vapor and atmospheric effects), 162 bands are remained in this data. There are four *endmembers* in this data: #1 Asphalt, includes the road, the parking area and a few roofs; #2 Grass covers areas with a green appearance in Fig. 3a; #3 Tree, owning a different spectral signature with that of grass, appears dark green mainly in the bottom areas in Fig. 3a; #4 Roof, appears white in the center in Fig. 3a.



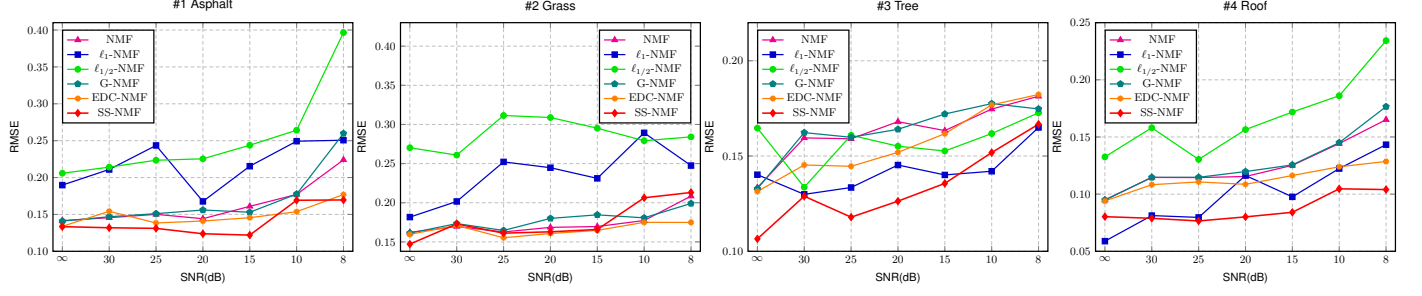


Figure 5. RMSEs of four *abundance* maps vs. SNRs on the Urban data: (a) Asphalt, (b) Grass, (c) Tree and (d) Roof. The symbol ‘ $\infty$ ’ at the bottom-left of each subfigure indicates that no Gaussian noise is added manually.

Table IV

THE AVERAGE RMSEs AND THEIR STANDARD DEVIATIONS VS. SNRS ON THE URBAN DATA. THE VALUES MARKED IN BOLD ARE THE BEST.

| SNR      | Root Mean Square Error <b>RMSE</b> ( $\times 10^{-2}$ ) |                 |                 |                   |                 |                 |                                   |                                   |
|----------|---|-----------------|-----------------|-------------------|-----------------|-----------------|-----------------------------------|-----------------------------------|
|          | VCA   | NMF             | $\ell_1$ -NMF   | $\ell_{1/2}$ -NMF | G-NMF           | W-NMF           | EDC-NMF                           | SS-NMF                            |
| $\infty$ | $27.6 \pm 3.62$   | $13.3 \pm 0.09$ | $14.3 \pm 0.07$ | $19.3 \pm 0.15$   | $13.2 \pm 0.07$ | $13.7 \pm 0.07$ | $13.0 \pm 0.06$                   | <b><math>11.7 \pm 0.06</math></b> |
| 30       | $30.5 \pm 0.48$   | $14.7 \pm 0.24$ | $15.6 \pm 0.42$ | $19.2 \pm 0.24$   | $14.9 \pm 0.21$ | $16.3 \pm 0.32$ | $14.5 \pm 0.38$                   | <b><math>12.8 \pm 0.07</math></b> |
| 25       | $30.5 \pm 0.57$   | $14.7 \pm 0.04$ | $17.7 \pm 0.05$ | $20.7 \pm 0.13$   | $14.8 \pm 0.05$ | $17.0 \pm 0.11$ | $13.7 \pm 0.04$                   | <b><math>12.2 \pm 0.06</math></b> |
| 20       | $30.6 \pm 0.54$   | $14.9 \pm 0.02$ | $16.9 \pm 1.59$ | $21.1 \pm 0.14$   | $15.5 \pm 0.05$ | $20.3 \pm 0.25$ | $14.1 \pm 0.05$                   | <b><math>12.3 \pm 0.07</math></b> |
| 15       | $30.6 \pm 0.70$   | $15.5 \pm 0.05$ | $17.1 \pm 0.74$ | $21.6 \pm 0.11$   | $15.9 \pm 0.03$ | $27.3 \pm 0.31$ | $14.7 \pm 0.04$                   | <b><math>12.7 \pm 0.07</math></b> |
| 10       | $30.7 \pm 0.92$   | $16.8 \pm 0.04$ | $20.1 \pm 0.04$ | $22.3 \pm 1.66$   | $17.0 \pm 0.04$ | $21.6 \pm 0.01$ | <b><math>15.7 \pm 0.03</math></b> | $15.8 \pm 0.06$                   |
| 8        | $30.7 \pm 0.57$   | $19.5 \pm 1.60$ | $20.2 \pm 1.09$ | $27.2 \pm 0.15$   | $20.3 \pm 0.51$ | $23.8 \pm 0.36$ | $16.6 \pm 0.04$                   | <b><math>16.3 \pm 0.04</math></b> |
| Avg.     | 30.2  | 15.6            | 17.4            | 21.6              | 15.9            | 20.0            | 14.6                              | <b>13.4</b>                       |

**Jasper Ridge** is a popular hyperspectral data set introduced in [51]. There are  $512 \times 614$  pixels in it. Each pixel is observed at 224 bands covering the wavelengths ranging from 380 *nm* to 2500 *nm*. The spectral resolution is 10 *nm*. As the original hyperspectral data set is too complex to get the Ground Truth, we consider a subimage with  $100 \times 100$  pixels. The first pixel corresponds to the pixel (105, 269) in the original image. After removing bands 1–3, 108–112, 154–166 and 220–224 (due to dense water vapor and atmospheric effects), we remain 198 bands (this is a common preprocess for the HU analysis). There are four *endmembers* latent in data: #1 Road, #2 Soil, #3 Water and #4 Tree, as shown in Fig. 3b.

The ground truths for both data sets are determined by using the method proposed by [52], [38]. There are three steps. First, the VD method, proposed by [53], is adopted to determine the endmember number. Second, the endmember spectra are manually chosen from hyperspectral data. These spectra have to enjoy high similarity with the reference hyperspectral spectra, which are supplied by the USGS mineral spectral library<sup>3</sup> etc. Given the *endmembers*, we can get the corresponding *abundances* by solving a constraint convex optimization problem, which can be easily implemented by using the Matlab Optimization Toolbox. The ground truths of the listed data sets are showed in the last column of Figs. 12, 13, 14 and 15.

<sup>3</sup>Available on <http://speclab.cr.usgs.gov/spectral-lib.html>.

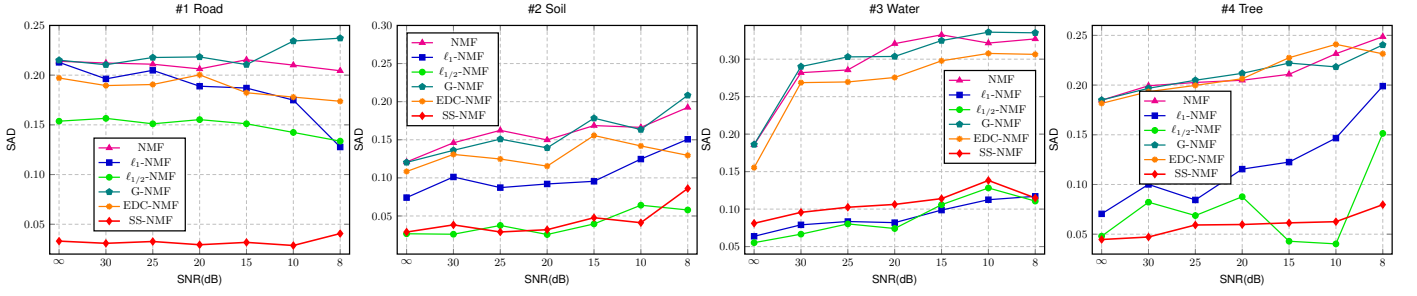


Figure 6. SADs of four *endmembers* vs. SNRs on the Jasper Ridge data: (a) Road, (b) Soil, (c) Water and (d) Tree. The symbol ‘ $\infty$ ’ at the bottom-left of each subfigure indicates that no Gaussian noise is added manually.

Table V

THE AVERAGE SADs AND THEIR STANDARD DEVIATIONS VS. SNRS ON THE JASPER RIDGE DATA. THE VALUES MARKED IN BOLD ARE THE BEST.

| SNR      | Spectral Angle Distance $\overline{\text{SAD}} (\times 10^{-2})$ |                 |                 |                   |                 |                 |                 |                                  |
|----------|--|-----------------|-----------------|-------------------|-----------------|-----------------|-----------------|----------------------------------|
|          | VCA  | NMF             | $\ell_1$ -NMF   | $\ell_{1/2}$ -NMF | G-NMF           | W-NMF           | EDC-NMF         | SS-NMF                           |
| $\infty$ | $34.7 \pm 1.38$  | $17.6 \pm 0.16$ | $10.5 \pm 0.08$ | $7.1 \pm 0.34$    | $17.7 \pm 0.16$ | $23.0 \pm 0.43$ | $16.1 \pm 0.10$ | <b><math>4.7 \pm 0.10</math></b> |
| 30       | $29.1 \pm 7.27$  | $21.0 \pm 0.14$ | $11.9 \pm 0.08$ | $8.3 \pm 0.21$    | $20.8 \pm 0.17$ | $20.9 \pm 0.30$ | $19.6 \pm 0.17$ | <b><math>5.3 \pm 0.12</math></b> |
| 25       | $32.0 \pm 7.55$  | $21.5 \pm 0.22$ | $11.5 \pm 0.10$ | $8.4 \pm 0.20$    | $21.9 \pm 0.23$ | $26.0 \pm 0.24$ | $19.6 \pm 0.14$ | <b><math>5.6 \pm 0.12</math></b> |
| 20       | $32.2 \pm 10.63$   | $22.0 \pm 0.19$ | $12.0 \pm 0.08$ | $8.6 \pm 0.18$    | $21.8 \pm 0.21$ | $28.9 \pm 0.19$ | $19.9 \pm 0.39$ | <b><math>5.7 \pm 0.09</math></b> |
| 15       | $37.8 \pm 9.83$  | $23.2 \pm 0.21$ | $12.6 \pm 0.14$ | $8.5 \pm 0.21$    | $23.4 \pm 0.22$ | $31.7 \pm 0.21$ | $21.6 \pm 0.24$ | <b><math>6.4 \pm 0.12</math></b> |
| 10       | $42.8 \pm 12.77$   | $23.2 \pm 0.22$ | $14.0 \pm 0.12$ | $9.4 \pm 0.35$    | $23.8 \pm 0.20$ | $37.1 \pm 0.09$ | $21.7 \pm 0.43$ | <b><math>6.8 \pm 0.11</math></b> |
| 8        | $45.2 \pm 12.53$   | $24.3 \pm 0.23$ | $14.8 \pm 0.13$ | $11.3 \pm 1.02$   | $25.5 \pm 0.26$ | $39.6 \pm 0.12$ | $21.0 \pm 0.06$ | <b><math>8.0 \pm 0.25</math></b> |
| Avg.     | 36.3   | 21.8            | 12.5            | 8.8               | 22.1            | 29.6            | 19.9            | <b>6.1</b>                       |

### E. Performance Evaluation

To test the robustness, this subsection evaluates the influence of noise on the HU performances for all methods. We choose the i.i.d. zero-mean white Gaussian noise rather than the correlated noise for two reasons. First, without any prior, the correlated noise is very hard to add to the hyperspectral image. Second, the zero-mean white Gaussian noise is the most widely used noise in the HU study [25], [26], [49].

For each hyperspectral data set, seven experiments are carried out with respect to seven levels of Gaussian noise (i.e., SNR =  $\infty$ , 30, 25, 20, 15, 10, 8 dB). Each experiment is repeated 50 times, and the mean results as well as their corresponding standard deviations are provided here. The evaluation is organized in two parts: Quantitative Results and Qualitative Results.

1) *Quantitative Results and Analysis* : Figs. 4, 5 and Tables III, IV show the experiment results on the Urban data set. Figs. 4 and 5 illustrate the plots of the SAD (metric for estimated *endmembers*) and the RMSE (metric for estimated *abundances*) versus seven SNR levels of Gaussian noise respectively. The results of our proposed SS-NMF algorithm are consistently better than or comparable to all the other methods. Especially, SS-NMF achieves significantly better performances in Figs. 4b, 4c and 4d.

Tables III and IV show the detailed SAD and RMSE performances and their corresponding standard deviations versus seven levels of Gaussian noise. The values in Table III are the average of the statistics plotted in four subfigures in Fig. 4. Taking the value “5.7” marked in bold at the northeast corner of Table III for example, it is the average of the four SAD values of the SS-NMF algorithm under the ‘ $\infty$ ’ noise level condition in Fig. 4. Similarly, Table IV

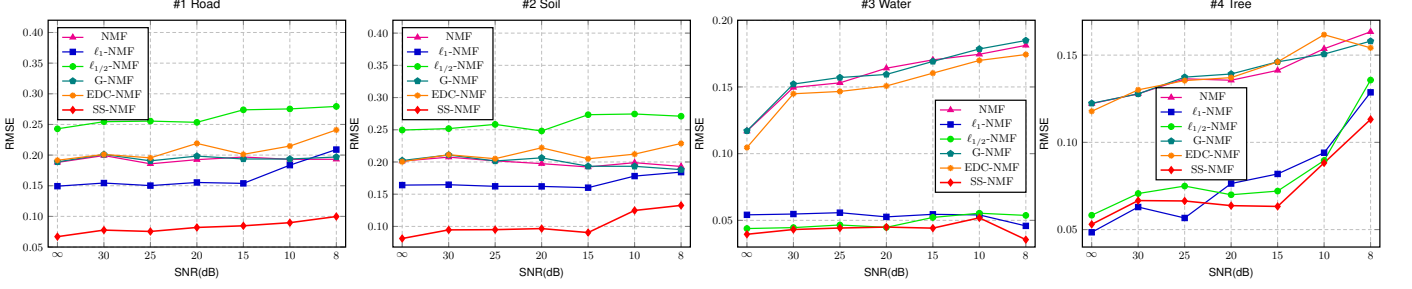


Figure 7. RMSEs of four *abundance* maps vs. SNRs on the Jasper Ridge data: (a) Road, (b) Soil, (c) Water and (d) Tree. The symbol ‘ $\infty$ ’ at the bottom-left of each subfigure indicates that no Gaussian noise is added manually.

Table VI

THE AVERAGE RMSEs AND THEIR STANDARD DEVIATIONS VS. SNRS ON JASPER RIDGE DATA. THE VALUES MARKED IN BOLD ARE THE BEST.

| SNR      | Root Mean Square Error $\overline{\text{RMSE}} (\times 10^{-2})$ |                 |                 |                   |                 |                 |                 |                                  |
|----------|--|-----------------|-----------------|-------------------|-----------------|-----------------|-----------------|----------------------------------|
|          | VCA  | NMF             | $\ell_1$ -NMF   | $\ell_{1/2}$ -NMF | G-NMF           | W-NMF           | EDC-NMF         | SS-NMF                           |
| $\infty$ | $22.3 \pm 1.87$  | $15.7 \pm 0.18$ | $10.4 \pm 0.33$ | $14.9 \pm 0.21$   | $15.8 \pm 0.17$ | $15.6 \pm 0.24$ | $15.4 \pm 0.20$ | <b><math>6.0 \pm 0.24</math></b> |
| 30       | $23.7 \pm 6.32$  | $17.1 \pm 0.19$ | $10.9 \pm 0.32$ | $15.5 \pm 0.20$   | $17.3 \pm 0.24$ | $16.7 \pm 0.14$ | $17.2 \pm 0.45$ | <b><math>7.1 \pm 0.21</math></b> |
| 25       | $25.7 \pm 5.44$  | $16.9 \pm 0.13$ | $10.6 \pm 0.40$ | $15.9 \pm 0.20$   | $17.2 \pm 0.14$ | $17.8 \pm 0.19$ | $17.1 \pm 0.10$ | <b><math>7.0 \pm 0.29</math></b> |
| 20       | $25.8 \pm 7.53$  | $17.2 \pm 0.15$ | $11.2 \pm 0.22$ | $15.4 \pm 0.17$   | $17.6 \pm 0.18$ | $19.0 \pm 0.13$ | $18.2 \pm 0.62$ | <b><math>7.2 \pm 0.25</math></b> |
| 15       | $29.1 \pm 6.59$  | $17.5 \pm 0.15$ | $11.3 \pm 0.26$ | $16.8 \pm 0.18$   | $17.6 \pm 0.12$ | $20.9 \pm 0.11$ | $17.8 \pm 0.48$ | <b><math>7.1 \pm 0.11</math></b> |
| 10       | $32.1 \pm 7.95$  | $18.0 \pm 0.13$ | $12.7 \pm 0.40$ | $17.4 \pm 0.28$   | $17.9 \pm 0.17$ | $21.9 \pm 0.05$ | $19.0 \pm 0.73$ | <b><math>8.9 \pm 0.28</math></b> |
| 8        | $33.2 \pm 7.42$  | $18.3 \pm 0.12$ | $14.2 \pm 0.30$ | $18.5 \pm 0.68$   | $18.2 \pm 0.10$ | $23.6 \pm 0.03$ | $20.0 \pm 0.13$ | <b><math>9.5 \pm 0.31</math></b> |
| Avg.     | 27.4   | 17.3            | 11.6            | 16.3              | 17.4            | 19.4            | 17.8            | <b>7.5</b>                       |

illustrates the average RMSE values in four subfigures in Fig. 5. From the above results, we observe that

- For all the seven noise levels, our SS-NMF gets the best or comparable performances. Comparing to the best algorithms other than our proposed SS-NMF algorithm, i.e.  $\ell_{1/2}$ -NMF for SAD and EDC-NMF for RMSE respectively, SS-NMF achieves 59.12% decrement for SAD, as shown in Table III, and 8.22% decrement for RMSE, as shown in Table IV. Moreover, when the level of noise varies from SNR= $\infty$  to SNR=8, our method is the most stable one. Those all prove that the structured sparse regularization is well suited for the HU problem.
- Comparing with the results obtained by G-NMF [39] and W-NMF [48], the sparse regularized NMF methods (i.e.,  $\ell_1$ -NMF [47],  $\ell_{1/2}$ -NMF [25]) achieve better or comparable results. This demonstrates that the sparse constraint is more important than the graph constraint when estimating *endmembers*. This is mainly because the graph constraint helps to find a new *abundance* space, in which similar pixels share similar *abundances*. Hence it leads to smooth or hazy *abundances*, which is closely related to less expressive [16] or inaccurate *endmembers* by the iterative updating rules introduced by [13], [39].
- NMF and its extensions (SS-NMF and  $\ell_1$ -NMF [47],  $\ell_{1/2}$ -NMF [25], G-NMF [39]) outperform the VCA algorithm significantly. There are two reasons for this: (1) NMF based methods overcome the requirement of pure pixels, which is essential for the VCA method. (2) The parts-based property of NMF helps to discover latent *endmembers* and their corresponding *abundances*.

Figs. 6, 7 and Tables V, VI show the experiment results on the Jasper Ridge data. The HU results and their standard

deviations versus seven levels of noise are summarized in Tables V, VI. Graphical plots are shown in Figs. 6, 7. As we can see, our proposed SS-NMF are consistently better than or comparable to all the other methods. In particular, Figs. 6a, 7a, 7b and Tables V, VI show that SS-NMF outperform the other algorithms significantly. Comparing to the best algorithm other than our proposed algorithm, i.e.  $\ell_{1/2}$ -NMF for SAD and  $\ell_1$ -NMF for RMSE respectively, SS-NMF achieves 30.68% decrement for SAD (metric for *endmember*) in Table V and 35.34% decrement for RMSE (metric for *abundance*) in Table VI.

2) *Qualitative Results and Analysis*: In order to give an intuitive comparison of the HU results, we illustrate the *abundance* maps on Urban data in Figs. 12, 13 and the *abundance* maps on Jasper Ridge data in Figs. 14, 15. There are two ways to show the *abundance* maps: in pseudo color (as shown in Figs. 12, 13a, 14, and 15a) and in gray scale (as shown in Figs. 13b and 15b). Taking the last subfigure in Fig. 13a for example, we illustrate fractional *abundances*  $A_{kn}$  associated with pixel  $y_n$  by plotting the corresponding pixel using proportions of Red, Blue, Green and Black ink given by  $A_{kn}$  for  $k = 1, 2, 3, 4$ , respectively [54]. So, for instance, a pixel for which  $A_{1n} = 1$  will be colored red, whereas one for which  $A_{1n} = A_{2n} = 0.5$  will be colored with equal proportions of Red and Blue ink and so will appear Purple. The Figs. 12, 14, and 15a are achieved in this way. The gray scale way is straightforward. Taking the subfigures in the last column in Fig. 13b for example, we illustrate fractional *abundances*  $A_{kn}$  associated with pixel  $y_n$  by plotting the corresponding pixels in the four subfigures with different gray scale appearances given by  $A_{kn}$  for  $k = 1, 2, 3, 4$  respectively. So, for instance, a pixel for which  $A_{1n} = 1$  will be white in the first subfigure and black in the other three subfigures.

Comparing the results in Figs. 12, 13, 14 and 15, we have

- SS-NMF achieves the most similar *abundance* maps according to the Ground Truths. In addition, the colors in our results are regional smooth and sparsely mixed by Red, Blue, Green and Black ink. These demonstrate that the structured sparse constraint is effective and meaningful.
- *Abundance* maps estimated by the sparse regularized NMF methods ( $\ell_1$ -NMF,  $\ell_{1/2}$ -NMF) contain more noise than those of SS-NMF and G-NMF. The reason is that the graph structure constraint is a kind of smooth constraint, which could urge the learnt *abundance* maps to be smooth.

#### F. The Performance versus varying parameters

In this subsection, we evaluate the impact of varying regularized parameters upon the performances. Nine experiments are conducted with respect to nine different regularized parameters on the hyperspectral images, which are not degraded by manually adding Gaussian noise. The regularized parameters vary in the following way:  $\lambda = 2^{-4}\hat{\lambda}, 2^{-3}\hat{\lambda}, 2^{-2}\hat{\lambda}, 2^{-1}\hat{\lambda}, 2^0\hat{\lambda}, 2^1\hat{\lambda}, 2^2\hat{\lambda}, 2^3\hat{\lambda}, 2^4\hat{\lambda}$  and  $\alpha = 2^{-4}\hat{\alpha}, 2^{-3}\hat{\alpha}, 2^{-2}\hat{\alpha}, 2^{-1}\hat{\alpha}, 2^0\hat{\alpha}, 2^1\hat{\alpha}, 2^2\hat{\alpha}, 2^3\hat{\alpha}, 2^4\hat{\alpha}$ . Here the values of  $\hat{\lambda}, \hat{\alpha}$  are the ones that help SS-NMF to get the best results on each data set. Obviously, they are different with respect to the two data sets. To reduce randomness of the results, each experiment is repeated ten times and the mean results are provided.

Fig. 8 shows the graphical performances: (a) on Urban data, (b) on Jasper Ridge data. For simplicity, the average performances are provided for each hyperspectral image:  $\overline{\text{SAD}}$  on the left side and  $\overline{\text{RMSE}}$  on the right side. Taking the Fig. 8a for example, each value in the left subfigure is achieved by performing an average calculation of the SADs of all estimated *endmembers*; Each value in the right subfigure is obtained by averaging the RMSEs of all estimated *abundance* maps. There is no regularized parameter in NMF, leading to the graphical results of NMF to be plain. From the results in Fig. 8, we have

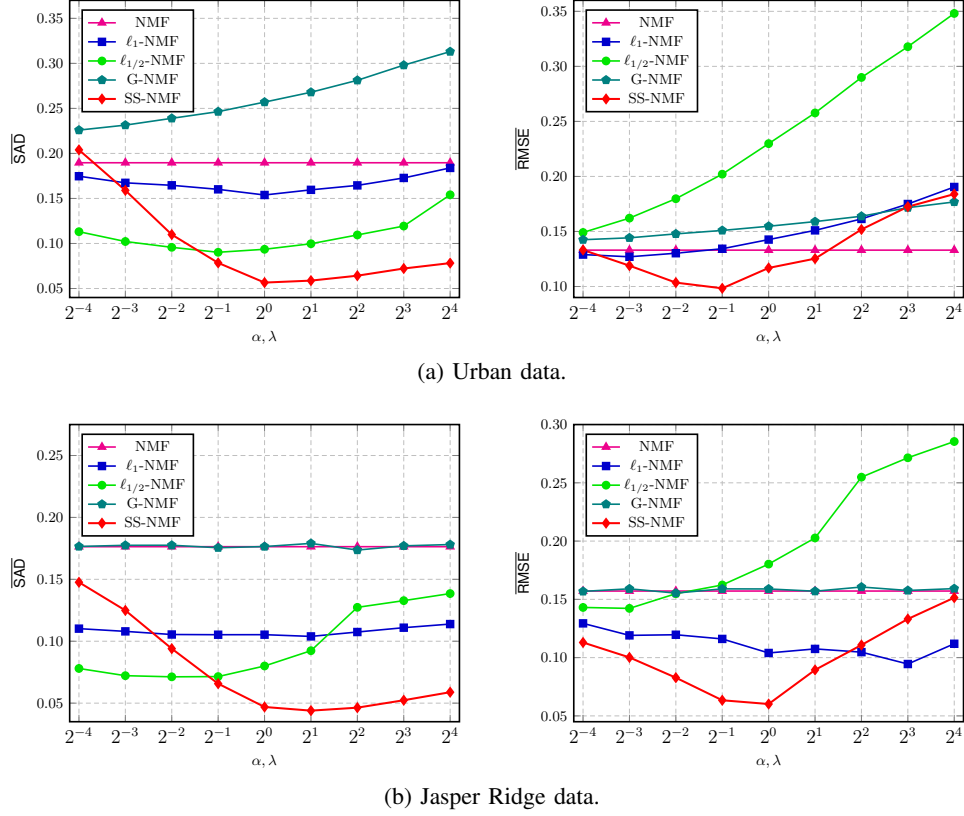


Figure 8. Performance vs. the varying parameters  $\lambda$  and  $\alpha$ : (a) on Urban data, (b) on Jasper Ridge data. The first column shows the  $\overline{\text{SAD}}$ s, while the second column displays the  $\text{RMSE}$ s. The symbol ‘ $2^4$ ’ at the bottom-right of each subfigure indicates  $\alpha, \lambda$  equal to  $2^4$  times of the values of the best parameter setting respectively.

- As the regularized parameters  $\{\lambda, \alpha\}$  vary from  $2^{-4} \{\hat{\lambda}, \hat{\alpha}\}$  to  $2^4 \{\hat{\lambda}, \hat{\alpha}\}$ , the performances of SS-NMF increase first and then decrease. The proposed SS-NMF gets good performance in the interval of  $\lambda = [2^{-2}, 2^2] \hat{\lambda}$ ,  $\alpha = [2^{-2}, 2^2] \hat{\alpha}$ . Especially, SS-NMF outperforms the other algorithms greatly under suitable parameter setting  $\lambda = \hat{\lambda}, \alpha = \hat{\alpha}$ .
- It seems  $\ell_1$ -NMF,  $\ell_{1/2}$ -NMF and G-NMF are more robust to the varying regularized parameters than our SS-NMF method. There might be two reasons for this phenomenon. First, the results of  $\ell_1$ -NMF,  $\ell_{1/2}$ -NMF and G-NMF are much worse than that of our method. That is, the varying regularized parameters cannot make these algorithms to get much better or much worse results under different parameter conditions. Second, although our SS-NMF method could achieve significantly better results than the other methods, it achieves bad results under extreme parameter settings.

### G. Influences of Weighting Methods

There are many weighting methods. Three common ones are considering the feature distance only, the spatial distance only and the spatial and feature distances simultaneously. The second and the third methods are well suited for hyperspectral images. The reason is that hyperspectral images enjoy the geometrically intuitive property, that is, the pixels are located on the grids of a 2-D plane, which means the spatial neighborhood is inherent for each pixel. In the previous experiments, we employ the third weighting method, which treats the pixels among the 30% biggest SADs in the local windows as nearest neighbors. In this subsection, the HU performances of the second and the



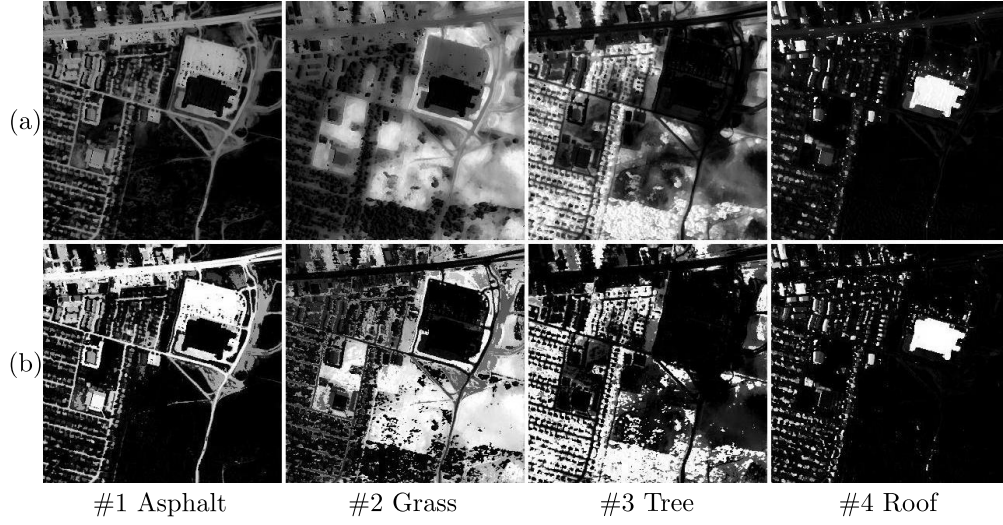


Figure 9. The performances of SS-NMF vs. two weighting methods on Urban data. (a) Only the spatial distance is considered when the local graph structures are constructed. (b) Both spatial and feature distances are considered when the local structures are constructed. Each column shows one kind of *abundance* maps respectively, i.e. #1 Asphalt, #2 Grass, #3 Tree and #4 Roof.

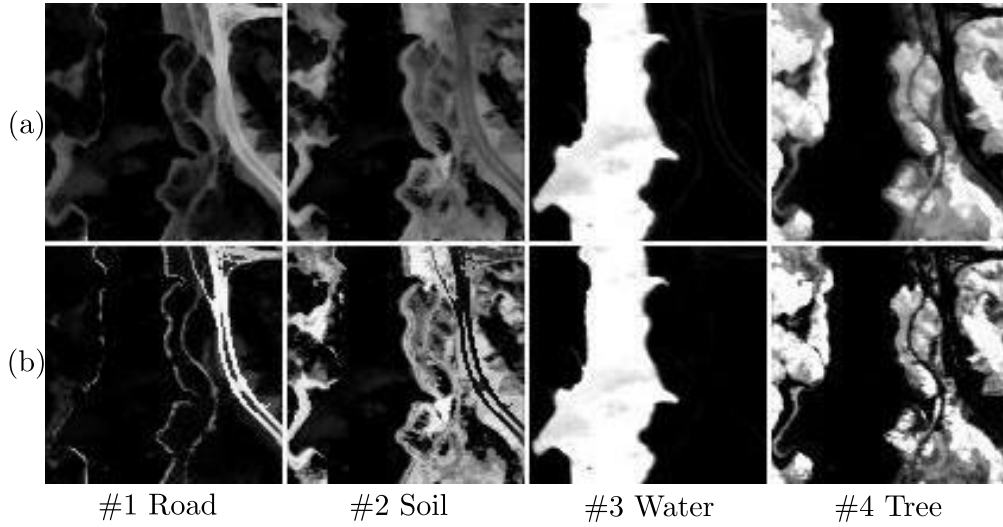


Figure 10. The performances of SS-NMF vs. two weighting methods on Jasper Ridge data. (a) Only the spatial distance is considered when the local graph structures are constructed. (b) Both spatial and feature distances are considered when the local structures are constructed. Each column shows one kind of *abundance* maps respectively, i.e. #1 Road, #2 Soil, #3 Water and #4 Tree.

third weighting methods are thoroughly compared.

Figs. 9 and 10 show the *abundance* maps obtained by the SS-NMF algorithm with different weighting methods: (a) considering the spatial distance only and (b) considering the spatial and feature distances simultaneously. This experiment is conducted on the hyperspectral images, which are not degraded by manually adding Gaussian noise. As Figs. 9a and 10a show, the results of (a) weighting method are too smooth and hazy. The reason is that only considering the spatial distance easily urges a lot of pixels with dissimilar spectral signatures to be connected on the graph. Accordingly, minimizing the objective function transfers these graph constraints to the *abundance* space. Besides, this part of improper constraints would confuse the optimization algorithm when solving the nonconvex problem, resulting in bad minima.



Table VII  
THE TIME (IN SECOND) OF CONSTRUCTING GRAPH & ITERATION & CONVERGENCE ON THE TWO DATA SETS.

| Time (s)                     | Urban data set   |                  | Jasper Ridge data set |                 |
|------------------------------|------------------|------------------|-----------------------|-----------------|
|                              | NMF              | SS-NMF           | NMF                   | SS-NMF          |
| $t_{\text{construct graph}}$ | –                | $5.45 \pm 0.01$  | –                     | $0.60 \pm 0.01$ |
| $t_{\text{iteration}}$       | $14.03 \pm 0.12$ | $8.83 \pm 0.11$  | $2.19 \pm 0.02$       | $1.39 \pm 0.13$ |
| $t_{\text{convergence}}$     | $14.03 \pm 0.12$ | $14.28 \pm 0.11$ | $2.19 \pm 0.02$       | $1.99 \pm 0.14$ |

### H. Convergence Study

The objective function for SS-NMF has been proved non-increasing under the updating rules (19) and (20). In this subsection, we study the convergence rate and convergence time for the proposed algorithm. Fig. 11 illustrates the convergence curves of NMF and SS-NMF on the two hyperspectral data sets: (a) on Urban data set and (b) on Jasper Ridge data set. In each subfigure, the  $x$ -axis shows the number of iterations and the  $y$ -axis displays the energies of the objective functions. As can be seen, the updating rules for both algorithms are efficient, usually within 50 iterations; SS-NMF converges a bit faster, converging within 30 iterations.

The convergence time, measured in seconds, for NMF and SS-NMF is summarized in Table VII. There are three rows. The first row shows the time of constructing weighted graphs. Since NMF does not need this part, the time of constructing graphs is zero for this method, which is denoted by “–” in the table. The second row shows the time of iteration; The total convergence time is summarized in the last row. Comparing the results in Table VII, we have

- The SS-NMF method costs less time than the NMF method does during the iteration process. The reason is that SS-NMF needs less iterations than NMF does before convergence, as shown in Fig. 11.
- Although NMF costs less convergence time than SS-NMF does on the Urban data, it fails to continue this superiority on the Jasper Ridge data. The reason might be that the size of Jasper Ridge data is much smaller than that of the urban data, approximately  $\frac{1}{9}$ . It leads to the ratio of  $\frac{t_{\text{construct graph}}}{t_{\text{convergence}}}$  to be smaller on the Jasper Ridge data than that on the Urban data, i.e., 0.30 and 0.38 respectively.

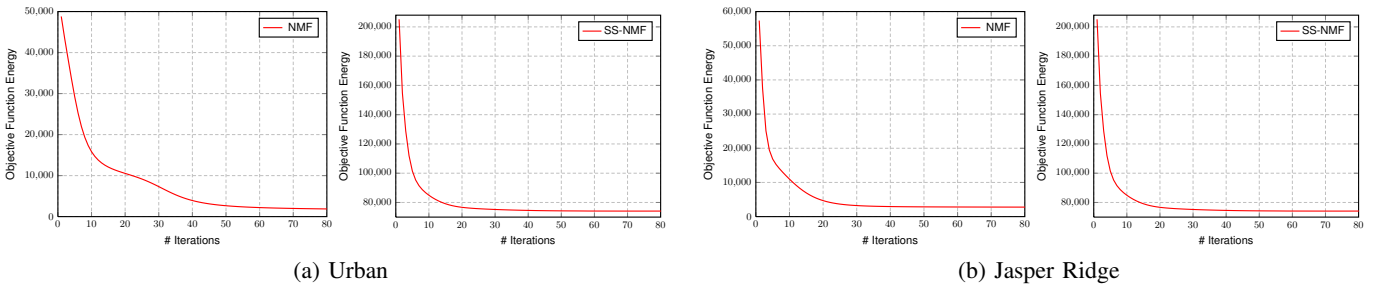


Figure 11. Convergence curves of NMF and SS-NMF.

## V. CONCLUSION

Based on the observed properties of hyperspectral data, we propose an effective method for the HU problem by imposing a Structured Sparse constraint in the NMF framework, which is called SS-NMF for short. SS-NMF can effectively overcome the three limitations of NMF for HU problem. With the structured sparse constraint, SS-NMF not only transfers the manifold structures inherently embedded in the original data space to the learned *abundance*



space, but also learns an expressive and accurate set of *endmembers*. Experiments on several hyperspectral data sets show that our method achieves better results than the state-of-the-art methods in the sense of both quantitative and qualitative performances. Besides, our method is relatively robust to different noise levels.

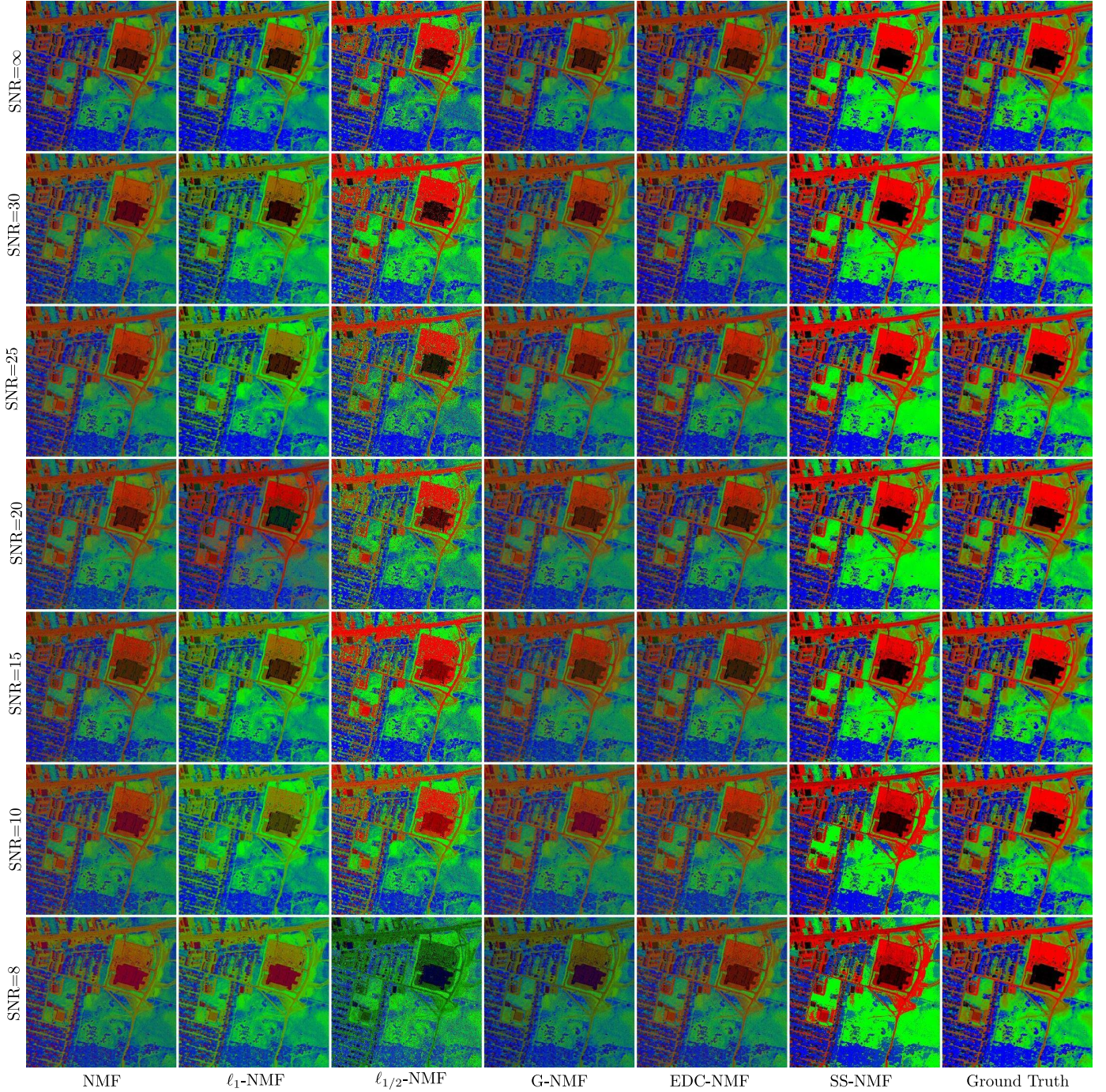


Figure 12. *Abundance* maps vs. seven noise levels (SNR=  $\infty$ , 30, 25, 20, 15, 10, 8 dB) on the Urban data. There are seven rows and seven columns in this figure. Each row displays the results under one noise level. From the 1-st to the 6-th column, each column illustrates the results for one algorithm. The last column shows the Ground Truths. For each subfigure, the proportions of Red, Blue, Green and Black ink associated with each pixel represent the fractional *abundances* of Asphalt, Tree, Grass and Roof in the corresponding pixel. This figure is best viewed in color.



## REFERENCES

- [1] N. Keshava, "A survey of spectral unmixing algorithms," *Lincoln Laboratory Journal*, vol. 14, no. 1, pp. 55–78, Jan 2003.
- [2] L. Tits, W. D. Keersmaecker, B. Somers, G. P. Asner, J. Farifteh, and P. Coppin, "Hyperspectral shape-based unmixing to improve intra- and interclass variability for forest and agro-ecosystem monitoring," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 74, no. 0, pp. 163 – 174, 2012.
- [3] N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE Transactions on Signal Processing (TSP)*, vol. 19, no. 1, pp. 44–57, 2002.
- [4] Bioucas-Dias *et al.*, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 2, pp. 354 –379, april 2012.
- [5] J. M. P. Nascimento and J. M. B. Dias, "Vertex component analysis: a fast algorithm to unmix hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 43, no. 4, pp. 898–910, 2005.
- [6] C.-I. Chang, C.-C. Wu, W. Liu, and Y. C. Ouyang, "A new growing method for simplex-based endmember extraction algorithm," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 44, no. 10, pp. 2804–2819, 2006.
- [7] G. Martin and A. Plaza, "Spatial-spectral preprocessing prior to endmember identification and unmixing of remotely sensed hyperspectral data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 2, pp. 380–395, 2012.
- [8] N. Dobigeon, S. Moussaoui, M. Coulon, J.-Y. Tourneret, and A. O. Hero, "Joint bayesian endmember extraction and linear unmixing for hyperspectral imagery," *IEEE Transactions on Signal Process (TSP)*, vol. 57, no. 11, pp. 4355–4368, 2009.
- [9] J. Wang and C.-I. Chang, "Applications of independent component analysis in endmember extraction and abundance quantification for hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 44, no. 9, pp. 2601–2616, 2006.
- [10] J. M. Bioucas-Dias, "A variable splitting augmented lagrangian approach to linear spectral unmixing," in *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, 2009, pp. 1–4.
- [11] J. M. P. Nascimento and J. M. Bioucas-Dias, "Hyperspectral unmixing based on mixtures of dirichlet components," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 50, no. 3, pp. 863–878, 2012.
- [12] M. E. Winter, "N-findr: an algorithm for fast autonomous spectral end-member determination in hyperspectral data," in *SPIE Conference Imaging Spectrometry*, 1999, pp. 266–275.
- [13] D. D. Lee and H. S. Seung, "Learning the parts of objects with nonnegative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, Oct 1999.
- [14] S. E. Palmer, "Hierarchical structure in perceptual representation," *Elsevier Journal of Cognitive Psychology*, vol. 9, no. 4, pp. 441 – 474, 1977.
- [15] N. K. Logothetis and D. L. Sheinberg, "Visual object recognition," *Annual Review of Neuroscience*, vol. 19, no. 1, pp. 577–621, 1996.
- [16] S. Z. Li, X. Hou, H. Zhang, and Q. Cheng, "Learning spatially localized, parts-based representation," in *IEEE International Conference on Computer Vision (CVPR)*, 2001, pp. 207–212.
- [17] R. Sandler *et al.*, "Nonnegative matrix factorization with earth mover's distance metric for image analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 33, no. 8, pp. 1590–1602, 2011.
- [18] W. Xu, X. Liu, and Y. Gong, "Document clustering based on non-negative matrix factorization," in *International Conference on Research and Development in Information Retrieval (SIGIR)*, 2003, pp. 267–273.
- [19] F. Shahnaz, M. W. Berry, V. P. Pauca, and R. J. Plemmons, "Document clustering using nonnegative matrix factorization," *Elsevier Journal Information Processing & Management*, vol. 42, no. 2, pp. 373–386, 2006.
- [20] A. D. Pascual-Montano *et al.*, "Nonsmooth nonnegative matrix factorization (nsnmf)," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 28, no. 3, pp. 403–415, march 2006.
- [21] L. Miao, H. Qi, and H. Qi, "Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 45, no. 3, pp. 765–777, 2007.
- [22] A. Huck and M. Guillaume, "Robust hyperspectral data unmixing with spatial and spectral regularized nmf," pp. 1–4, 2010.
- [23] X. Lu, H. Wu, Y. Yuan, P. Yan, and X. Li, "Manifold regularized sparse nmf for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 51, no. 5, pp. 2815–2826, 2013.
- [24] M.-D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 49, no. 6, pp. 2014–2039, 2011.
- [25] Y. Qian, S. Jia, J. Zhou, and A. Robles-Kelly, "Hyperspectral unmixing via sparsity-constrained nonnegative matrix factorization," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 49, no. 11, pp. 4282 –4297, nov 2011.
- [26] X. Liu, W. Xia, B. Wang, and L. Zhang, "An approach based on constrained nonnegative matrix factorization to unmix hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 49, no. 2, pp. 757–772, 2011.

- [27] X. Chen, S. Kim, Q. Lin, J. G. Carbonell, and E. P. Xing, "Graph-structured multitask regression and an efficient optimization method for general fused lasso," *Computing Research Repository (CORR)*, vol. abs/1005, no. 3, pp. 1–21, 2010.
- [28] X. Geng, Z. Xiao, L. Ji, Y. Zhao, and F. Wang, "A gaussian elimination based fast endmember extraction algorithm for hyperspectral imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 79, no. 0, pp. 211 – 218, 2013.
- [29] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2000, pp. 556–562.
- [30] H. Kim and H. Park, "Nonnegative matrix factorization based on alternating nonnegativity constrained least squares and active set method," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 2, pp. 713–730, 2008.
- [31] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, "Algorithms and applications for approximate nonnegative matrix factorization," *Computational Statistics & Data Analysis*, vol. 30, no. 1, pp. 155–173, 2007.
- [32] C.-J. Lin, "Projected gradient methods for nonnegative matrix factorization," *MIT Press Neural Computation*, vol. 19, no. 10, pp. 2756–2779, 2007.
- [33] A. Gersho and R. M. Gray, *Vector quantization and signal compression*. Norwell, MA, USA: Kluwer Academic Publishers, 1991.
- [34] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. New York, USA: Springer, 2002.
- [35] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2001, pp. 585–591.
- [36] F. R. K. Chung, *Spectral Graph Theory*. American Mathematical Society, Dec. 1996.
- [37] S. Xiang, F. Nie, and C. Zhang, "Semi-supervised classification via local spline regression," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 32, no. 11, pp. 2039–2053, 2010.
- [38] S. Jia and Y. Qian, "Constrained nonnegative matrix factorization for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 47, no. 1, pp. 161–173, 2009.
- [39] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 33, no. 8, pp. 1548 –1560, aug 2011.
- [40] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society*, vol. 58, no. 1, pp. 267–288, 1996.
- [41] D. L. Donoho, "Compressed sensing," *IEEE Transactions Information Theory*, vol. 52, no. 4, pp. 1289–1306, 1996.
- [42] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–38, 1977.
- [43] L. Saul and F. Pereira, "Aggregate and mixed-order markov models for statistical language processing," in *Conference on Empirical Methods in Natural Language Process*, 1997, pp. 81–89.
- [44] J. Kivinen and M. K. Warmuth, "Additive versus exponentiated gradient updates for linear prediction," *Journal of Information and Computational Science*, vol. 132, no. 1, pp. 1–64, 1997.
- [45] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms (Second Edition)*. MIT Press, 2001.
- [46] D. C. Heinz and C.-I. Chang, "Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 39, no. 3, pp. 529–545, 2001.
- [47] P. O. Hoyer, "Non-negative sparse coding," in *IEEE Workshop Neural Networks for Signal Processing*, 2002, pp. 557–565.
- [48] J. Liu, J. Zhang, Y. Gao, C. Zhang, and Z. Li, "Enhancing spectral unmixing by local neighborhood weights," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 5, pp. 1545–1552, 2012.
- [49] N. Wang, B. Du, and L. Zhang, "An endmember dissimilarity constrained non-negative matrix factorization method for hyperspectral unmixing," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 6, no. 2, pp. 554–569, 2013.
- [50] P. O. Hoyer and P. Dayan, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research (JML)*, vol. 5, no. 12, pp. 1457–1469, 2004.
- [51] Envi-Tutorials, "Envi classic vegetation hyperspectral analysis. <http://www.exelisvis.com/Learn/WhitepapersDetail/TabId/802/PID/2627/TagID/129/TagName/hyperspectralimagery/Default.aspx>," 2013.
- [52] S. Jia and Y. Qian, "Spectral and spatial complexity-based hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 45, no. 12, pp. 3867–3879, 2007.
- [53] C.-I. Chang and Q. Du, "Estimation of number of spectrally distinct signal sources in hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 42, no. 3, pp. 608–619, 2004.
- [54] C. M. Bishop, *Pattern Recognition and Machine Learning*, ser. Information Science and Statistics. Springer, 2006.



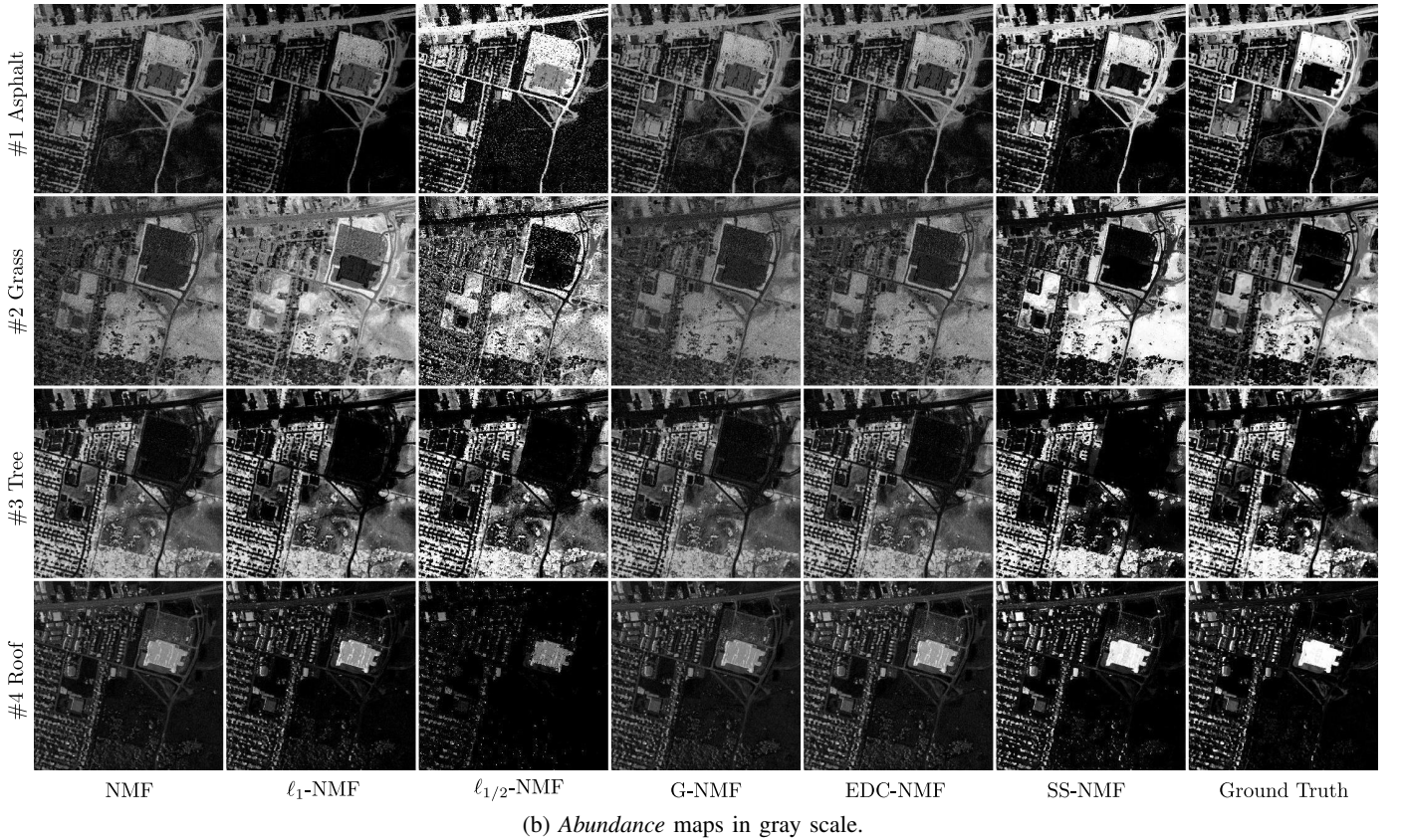
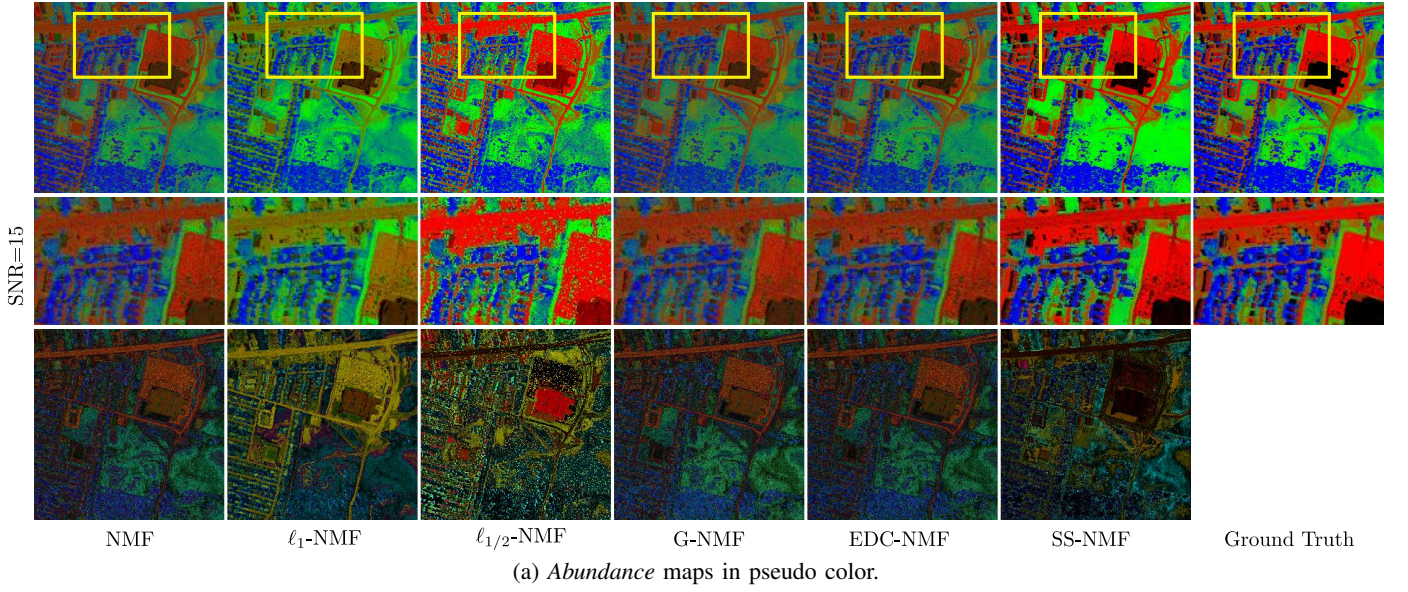


Figure 13. *Abundance maps under SNR= 15 dB noise level on the Urban data: (a) in pseudo color and (b) in gray scale. There are three rows in (a). The second row displays the results in the yellow box in the top row. The third row shows the absolute value of the difference between the estimated results and the ground truth. For each subfigure in (a), the proportions of Red, Blue, Green and Black ink associated with each pixel represent the fractional abundances of Asphalt, Tree, Grass and Roof in the corresponding pixels. There are four rows and seven columns in (b). Each row shows the abundance maps of one target. From the 1-st to the 6-th column, each column illustrates the results of one algorithm. The last column shows the Ground Truths. This figure is best viewed in color.*



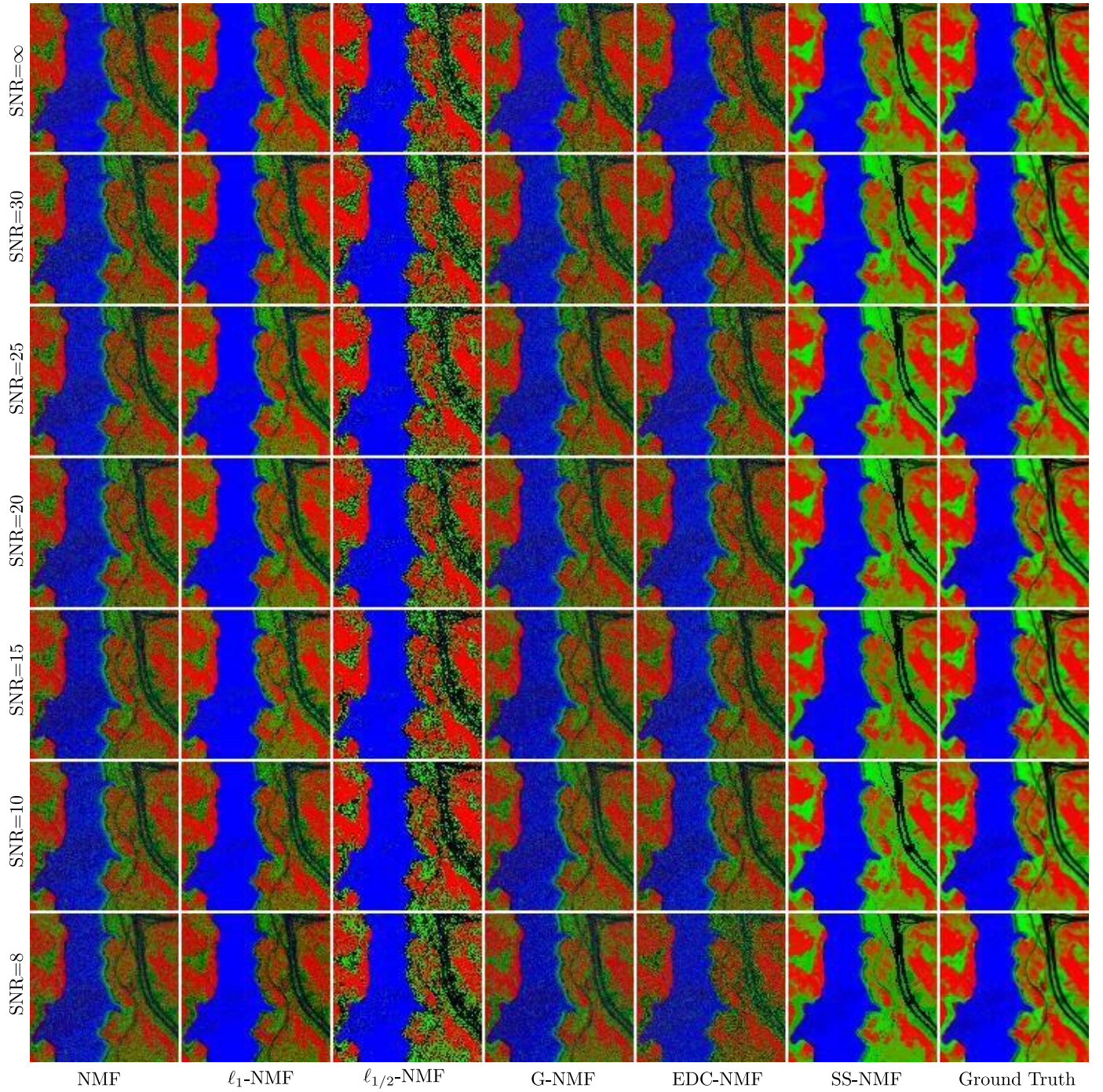


Figure 14. *Abundance* maps vs. seven noise levels (SNR=  $\infty$ , 30, 25, 20, 15, 10, 8 dB) on the Jasper Ridge data. There are seven rows and seven columns in this figure. Each row displays the results under one noise level. From the 1-st to the 6-th column, each column shows the results of one method. The last column illustrates the Ground Truths. For each subfigure, the proportions of Red, Blue, Green and Black ink associated with each pixel represent the fractional *abundances* of Tree, Water, Soil and Road in the corresponding pixel. This figure is best viewed in color.



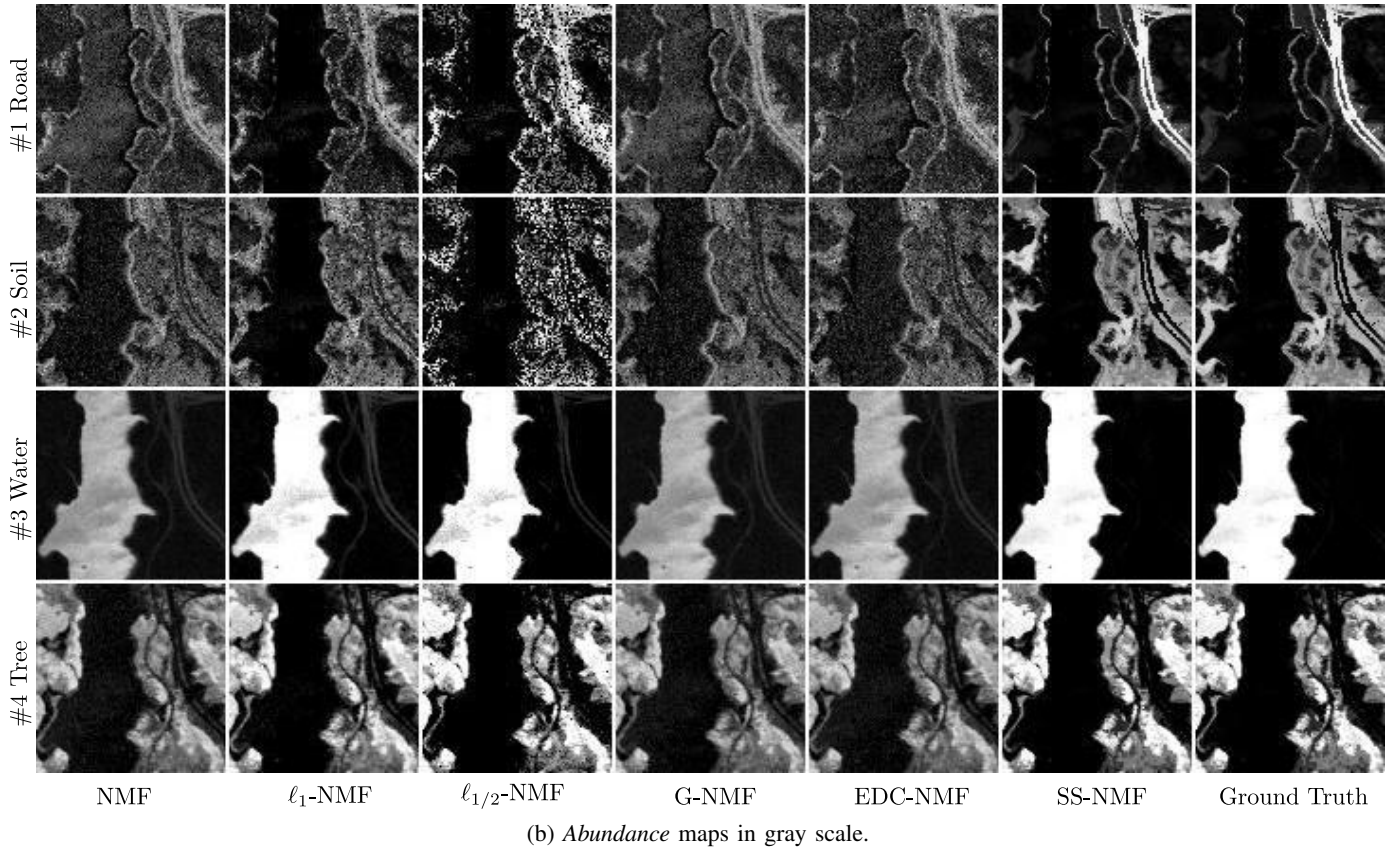
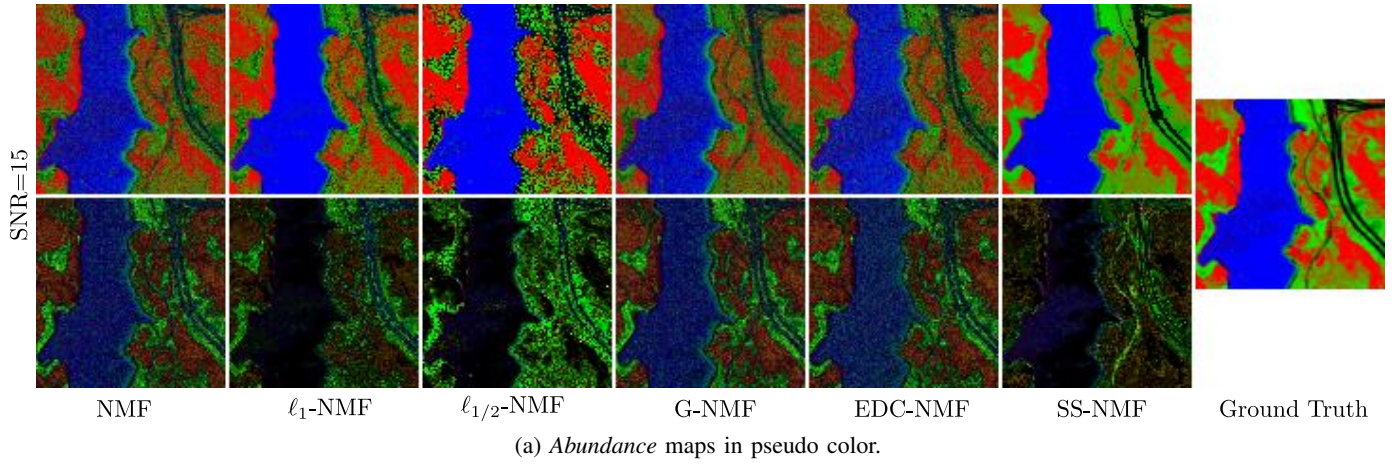


Figure 15. *Abundance maps under SNR= 15 dB noise level on the Japer Ridge data: (a) in pseudo color and (b) in gray scale. There are two rows in (a). The second row displays the absolute value of the difference between the estimated results and the ground truth. For each subfigure in (a), the proportions of Red, Blue, Green and Black ink associated with each pixel represent the fractional *abundances* of Tree, Water, Soil and Road in the corresponding pixel. There are four rows and seven columns in (b). Each row shows the *abundance* maps of one target. From the 1-st to the 6-th column, each column illustrates the results for one algorithm. The last column shows the Ground Truths. “#1 Road” in the first row and “#2 Soil” in the second row in (b) are two difficult targets due to the similarity spectral curves between them.*